# DESIGN OF MAC LAYER PROTOCOL FOR WIRELESS AD-HOC NETWORKS

By

**Bharti**

DEPARTMENT OF ELECTRICAL ENGINEERING

Indian Institute of Technology Kanpur

FEBRUARY, 2003

# DESIGN OF MAC LAYER PROTOCOL FOR WIRELESS AD-HOC NETWORKS

*A Thesis Submitted*
In Partial Fulfilment of the Requirements
for the Degree of
Master of Technology

by

BHARTI

Y110477

*to the*

DEPARTMENT OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

Feb, 2003

2 JUN 2003
पुरूषोत्तम काशीनाथ केलकर पुस्तकालय
भारतीय [illegible] कानपुर
अवाप्ति क्र० A 13485

POST GRADUATE OFFICE
THESIS SUBMITTED
ON 28-2-03
I.I.T. KANPUR

# CERTIFICATE

It is certified that the work contained in the thesis titled "*Design of MAC layer protocol for Wireless AD-HOC networks*", by Bharti, has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

(Dr Y N Singh)

Department Of Electrical Engineering

Indian Institute of Technology, Kanpur

27 Feb,2003

# ABSTRACT

**Name of student: Bharti, Roll No. Y110477**

**Master of Technology**

**Department of Electrical Engineering, IIT, Kanpur**

**Thesis Title: DESIGN OF MAC LAYER PROTOCOL FOR WIRELESS AD-HOC NETWORKS**

**Thesis Supervisor: Dr. Y.N. Singh**

**Feb. 2003**

Many medium-access control (MAC) protocols for wireless networks proposed or implemented to date are based on collision avoidance handshakes between sender and receiver In the vast majority of these protocols, including the IEEE 802 11 standard, the handshake is sender initiated, in that sender asks the receiver for permission to transmit using a short control packet, and transmits only after the receiver sends a short clear-to-send notification

We analyse the effect of making the collision-avoidance handshake; receiver initiated and compares the performance of a number of receiver-initiated protocols with the performance of sender-initiated collision avoidance protocols. But in the RIMA [11] and MACA-BI [10] the comparison of various protocols are not fairer, as MACA-BI indicates the higher throughput as compared to the other RIMA protocols and its various versions, while in all the versions of RIMA, it has shown, RIMA-DP as the best protocol among the receiver initiated policy. The heavy traffic approximation does not match the requirements of the multi-hop networks So the comparison of MACA-BI with RIMA protocols do not fit well. As from the discussion among the RIMA protocols, it is clear as we keep on increasing the number of nodes; the throughput variation in RIMA-BP is less as compared to other RIMA protocols. In this thesis, we have tried to show more variants of RIMA-BP protocols and its comparison with the original RIMA-BP protocol By considering some realistic assumptions, we have tried to show its effect on the throughput performance of various receiver initiated protocols

# Acknowledgement

I take this opportunity to express my sincere gratitude to my supervisor Dr. Y N Singh for his invaluable guidance. It would not have been possible for me to take this thesis to completion without his relentless support and encouragement. I consider myself extremely fortunate to have had a chance to work under his supervision I also wish to thank all the faculty members of the Department of Electrical Engineering for imparting their superb knowledge in course of my MTech program.

I also extend my thanks to the technical staff of the department for maintaining an excellent working facility. I would also like to thank my batch-mates those have made my stay in IIT Kanpur, the most memorable one. It is hard to forget the "bulla sessions" I used to have with my friends Asha, Alpanad, Dipti, Arpita, Apra, Jyoti, and bla bla in GH I want to express special thanks to Gandhi sir who helped and encouraged me during my work. I would like to thank my parents and brothers for providing me necessary support and encouragement for building a good career and a bright future. Finally, I thank the Almighty to keep showering me with all his love and luck and grant me an opportunity to be here in one of the world's best educational institution.

# CONTENTS

# LIST OF FIGURES AND TABLES

# LIST OF ABBREVIATIONS AND ACRONYMS

| | |
|---|---|
| MANET | Mobile AD-HOC Networks |
| MAC | Medium Access Control |
| CSMA | Carrier Sense Multiple Access |
| MACA | Medium Access Collision Avoidance |
| MACAW | Medium Access Collision Avoidance for Wireless networks |
| FAMA | Floor Acquistion Multiple Access |
| MACA-BI | MACA By Invitation |
| RIMA-SP | Receiver Initiated Multiple Access Simple Polling |
| RIMA-DP | Receiver Initiated Multiple Access Dual purpose Polling |
| RIMA-BP | Receiver Initiated Multiple Access Broadcast Polling |
| AP | Access Point |
| PC | Point Co-ordinator |
| MAN | Metropolitan Area Network |
| TP | Transmission Period |
| PCF | Point Co-ordination Function |
| DCF | Distributed Co-ordination Function |
| IFS | Inter Frame Space |
| NAV | Net Allocation Vector |
| RTS | Ready To Send |
| CTS | Clear To Send |
| RTR | Ready To Receive |
| CFP | Contention Free Period |
| CP | Contention Period |
| FCS | Frame Check Sequence |
| IBSS | Independent Basic Service Set |
| MPDU | MAC Protocol Data Unit |
| MSDU | MAC Service Data Unit |
| PHY | Physical Layer |
| CW | Contention Window |

# Chapter 1

# Introduction

In the next generation of wireless communication systems, there will be a need for the rapid deployment of independent mobile users Significant examples include establishing survivable, efficient, dynamic communication for emergency/rescue operations, disaster relief efforts, and military networks Such network scenarios cannot rely on centralised and organised connectivity, and can be conceived as applications of **Mobile Ad Hoc Networks.** A MANET is an autonomous collection of mobile users that communicate over relatively bandwidth constrained wireless links. Since the nodes are mobile, the network topology may change rapidly and unpredictably over time. The network is decentralised, where all network activity including discovering the topology and delivering messages must be executed by the nodes themselves, i e., routing functionality will be incorporated into mobile nodes

The set of applications for MANETs is diverse, ranging from small, static networks that are constrained by power sources, to large-scale, mobile, highly dynamic networks. The design of network protocols for these networks is a complex issue. Regardless of the application, MANETs need efficient distributed algorithms to determine network organisation, link scheduling, and routing However, determining viable routing paths and delivering messages in a decentralised environment where network topology fluctuates, is not a well-defined problem. While the shortest path (based on a given cost function) from a source to a destination in a static network is usually the optimal route, this idea is not easily extended to MANETs. Factors such as variable wireless link quality, propagation path loss, fading, multi-user interference, power spent, and topological changes, become relevant issues The network should be able to adaptively alter the routing paths to alleviate any of these effects

Moreover, in a military environment, preservation of security, latency, reliability, intentional jamming, and recovery from failure are significant concerns. Military networks are designed to maintain a low probability of intercept and/or a low probability of detection Hence, nodes prefer to radiate as little power as necessary and transmit as infrequently as possible, thus decreasing the probability of detection or interception. A lapse in any of these requirements may degrade the performance and dependability of the network

The design of MAC layer protocol is also a challenging issue in the wireless AD-HOC networks Earlier the protocols proposed for any environment were ALOHA and CSMA, but if the same were implemented in the wireless medium then the performance deterioration is enormous Hence to cop up with this situation, various sender and receiver initiated protocols were proposed

In sender initiated policy, the origination of the protocols started from MACA, which again gave iise to the development of MACAW, FAMA and IEEE802 11 As the prominent problems in the wireless AD-HOC networks are exposed and hidden nodes To deal with the hidden node problem, another concept for MAC layer protocols is incorporated i e receiver initiated policy.

In receiver initiated policy first variant which was derived from the old concept of MACA was MACA-BI As in the receiver initiated policy, receiver has to initiate the process of data transfer, hence the critical design issues in MAC protocols over a single channel are· (a) whether or not to use carrier sensing, (b) what persistent policy to use when transmitting or retransmitting packets, (c) how to resolve the collisions, and (d) how a receiver should poll its neighbours for data packets Keeping all these points in mind, a new variant of MACA-BI is proposed by J.J Garcia-Luna-Aceves et all called RIMA, which are having more versions as RIMA-SP, RIMA-BP and RIMA-DP. From the study of these protocols it is clear that as we keep on increasing the number of nodes the throughput performance is less effected in case of RIMA-BP over the other versions and also RIMA-DP was considered as the best receiver initiated protocol.

In this thesis, I have tried to give the offbeat of RIMA-BP in which the concept of waiting time is incorporated and this new variant comes out to be equivalent to the RIMA-DP with the less number of handshakes and less effect of hardware TX-to-RX time on its throughput performance at high channel rate.

This thesis is compiled into five chapters Chapter 1 is an introductory chapter briefing about the whole thesis work In Chapter 2, an overview about the various types of wireless networks, similarities and differences between the wired and wireless LANs, about the various infrastructure modes of wireless LANs and their specifications in the tabulated form are presented In Chapter 3 and 4, emphasise on various medium access techniques like static and dynamic channel allocation, is given. Last but not the least, in Chapter 5 the new medium access technique is proposed and its analytical and simulation results are discussed comprehensively

# Chapter 2

# Various Types of Wireless Networks

---

In this chapter, we have described various types of wireless networks and their specifications especially WLANs, which is the focus of interest for my thesis work Wireless local area networks are a star player in the wireless communications field, with growth projected at 100 percent per year for the next three years Users can deploy wireless LANs to transmit data, voice and video within individual buildings, across campuses, and over metropolitan areas Some of the computer and communications industries leading vendors are introducing Personal Digital Assistants (PDAs), modems, wireless microprocessors and other devices and applications in support of wireless communications

These are some of the new possibilities offered by Wireless LANs Alternative for adding new users to corporate LANs and supporting workers in remote locations, low cost alternative to cable-based systems, ubiquitous possibility (everytime and nearly everywhere) to access to any data base or any application located in the backbone [1]

## 2.1 Similarities and differences between wireless and wired LANs

There are many similarities and differences between wired LANs and wireless LANs.

### 2.1.1 Similarities between WLANs and wired LANs

Wireless LAN was designed to look and feel like any IEEE 802 wired LAN. It must support all of the protocols and all of the LAN management tools that operate on wired network To accomplish the task of similarity to wired LANs, IEEE 802 11 (wireless LAN standard) is designed to the same interface as IEEE 802.3. IEEE 802 11 operates under the IEEE802.2 logical link control (LLC) sublayer, providing all of the required services to support the LLC sublayer. In this way, WLAN is indistinguishable from IEEE 802.3 for the protocols that may be running above IEEE802.2

### 2.1.2 Differences between WLANs and wired LANs

There are also number of differences between wired and WLANs. The two most important differences are that (1) there are no wires (the air link) and (2) the mobility thus conferred by the lack of a wired tether. These differences lead to the tremendous benefits of a WLAN, as well as perceived drawback to them.

The air link is the radio or infrared link between WLAN transmitters and receivers Because WLAN transmissions are not confined to a wire, there may be concerns that the data carried by a WLAN is not private, not protected The data on a WLAN is broadcast for all to hear. Hence design of strong cryptographic mechanisms into the protocol is required foi the protection of data The airlink also exposes the transmissions of a wireless LAN to the vagaries of electromagnetic propagation For both radio and infraied based wireless LANs, everything in the environment is either a reflector or the attenuator of the signal carrying LAN data This can cause significant changes in the strength of a signal received by wireless LAN station, sometimes severing the station from the LAN entirely At the wavelengths used in the WLAN, small changes in position can cause large changes in the received signal strength This is due to the signal traveling via many seprate paths of differing length to arrive at the receiver

Each individual arriving signals is of a slightly different phase from that of all others. Adding these different phases together results in the composite received signal Since these individual signals sometime add up in phase and sometimes out of phase, the overall signal strength is sometimes large and sometimes small Objects moving in the environment, such as people, aluminized Mylar balloons, doors, and other objects, can also effect the strength of a signal at a receiver by changing the attenuation or reflection of the many individual signals

The second significant difference a WLAN has from a wired LAN is mobility The user of a WLAN is not tethered to the network outlet in the wall This is both the sources of the benefits of a WLAN and the cause of the internal complexity The benefit of mobility is that the LAN goes wherever you are, instantly and without the need to search for outlets or to arrange in advance with the network administrators In a laptop equipped with IEEE 802 11 WLAN connection, the connection is available in a coworker's office, down the hall in the conference room, downstairs in the lobby, across the parking lot in other building, even across the country on other campus This means that all of the information available over the network, while sitting in your office, is still available in all these locations· email, file servers, the company-internal web sites, and the Internet

Ofcourse, there is a flip side to the benefits of mobility Most of the network protocols and equipment in use today were not designed to cope with mobility. They were designed with the assumption that the addresses assigned to a network node would remain in a fixed location on the network. For example, early WLANs required that a mobile station could only roam within an area where the WLANs was connected to wired LAN, with only layer-2 bridges between the parts of the WLAN. This

requirement existed because there was no simple way to deal with the change of a layer-3 network address should the mobile station cross from one part of the network to another that is connected by a router. Today, there are ways to deal with this problem using new protocols, including DHCP and Mobile-IP

Another problem introduced by mobility is that location-based services lose their "hook" to a user's location, when network addresses are not locked to a physical location. Thus, notions such as the nearest network printer must be defined in a different way, when the physical location of a network user may be constantly changing This may increase the complexity of the service location provider, but meets the need of the mobile user

## 2.2 Types of Wireless LANs

Wireless LANs usually have two types of realization: "infrastructure" and "ad-hoc".

### 2.2.1 Ad Hoc LAN.

Several mobile nodes (e.g notebook computers) may get together in a small area (e.g in a conference room) and establish peer-to-peer communications among themselves without the help of *any infrastructure such as wired/wireless backbone* Since a small coverage area does not imply insured communication, there is no real reliability. As it will be explained in the section **Expected features of Wireless LANs**, the future standard should also support ad-hoc LANs Despite the possibility of ad-hoc networking, most applications will require communications with services located in a pre-existing infrastructure. Now question arises why to use ad-hoc LANs with so much complexities: Ad-hoc networks are advantageous in the way where setting up of fixed access points and backbone infrastructure is not always viable like infrastructure may not be present in a disaster area or war-zone; or infrastructure may not be practical for short range radios (Bluetooth range-10m).

#### 2.2.1.1 Concerns in building a Single hop Ad-hoc Network.

A wireless single hop ad-hoc network will have the following concerns:

1. *Throughput*: The air interface provides less capacity than a cable, hence there will be an efficiency issue in medium access protocol.
2. *Mobility*: The network should enable efficient user mobility, especially under a network without any base Station/AP (Access Point) for communication.

3 *Power consumption* Wireless devices are intended to be portable and mobile; and are typically battery powered, great attention has to be given to the power management

4 *Security.* In a wired network, the transmission medium can be physically secured, while a WLAN could be easily eavesdropped if not properly designed with some level of security Thus, using WLAN in a single hop ad-hoc network is probably a good, simple and feasible platform to cover most of these issues because of the these major factors (1)can satisfy the same typical requirements of any LAN, including high capacity, (2) full connectivity among attached stations and (3) broadcast capability, can meet some requirements specific to the wireless environment

**2.2.1.2 Hidden and Expose node problems in AD-HOC networks**

To see the nature of the problem, consider Fig. 2 1, where four wireless nodes are illustrated. The radio range is such that *A* and *B* are within each other's range and can potentially interfere with one another *C* can also potentially interfere with both *B* and *D*, but not with *A*

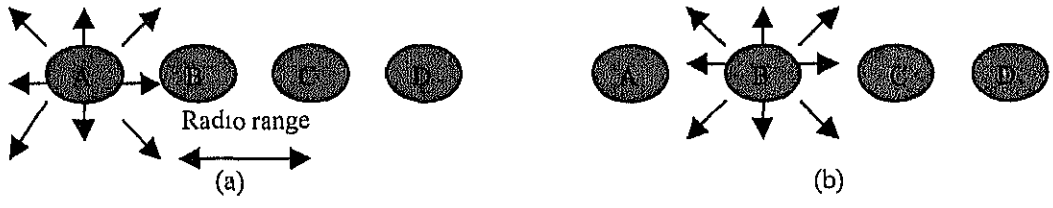



**Figure 2.1** A wireless LAN (a) *A* transmitting (b) *B* transmitting.

First consider what happens when *A* is transmitting to *B*, as depicted in fig. 2.1(a) If *C* senses the medium, it will not hear *A* because *A* is out of range, and thus falsely conclude that it can transmit. If *C* does start transmitting, it will interfere at *B*, wiping out the frame from *A*. The problem of a station not being able to detect a potential competitor for the medium because the competitor is too far away is called **hidden node problem.**

Now let us consider the reverse situation: *B* transmitting to *A*, as shown in Fig 2 1(b) If *C* senses the medium, it will hear an ongoing transmission and falsely conclude that it may not send to *D*, when in fact such a transmission would cause bad reception only in the zone between *B* and *C*, where neither of the intended receivers is located. This is called the **exposed node problem.**

## 2.2.2 Infrastructured wireless LANs.

Such an infrastructure is typically a higher-speed wired (or wireless) backbone Therefore, we can divide typical network traffic into two directions *uplink* (into the backbone) and *downlink* (from the backbone). The contact points to the backbone are called access points. The access points can be either base stations for wired infrastructures or wireless bridges for wireless infrastructures Repeaters may also be used for enlarging the coverage area of communication

**Downlink Traffic**: Due to the limited bandwidth of wireless LANs, a common channel is typically used for communication between an access point and mobile nodes. Downlink is achieved by *broadcasting* on this common channel. More precisely, the access point broadcasts packets to all mobile nodes even if there is only one destination Downlink activity may constitute up to 75 or 80 percent of the total traffic in wireless LANs because those nodes on modern LANs often operate in a client-server mode For instance there might be a high performance workstation or PC acting as a file server. A request for file transfer on the uplink may result in a huge file on the downlink

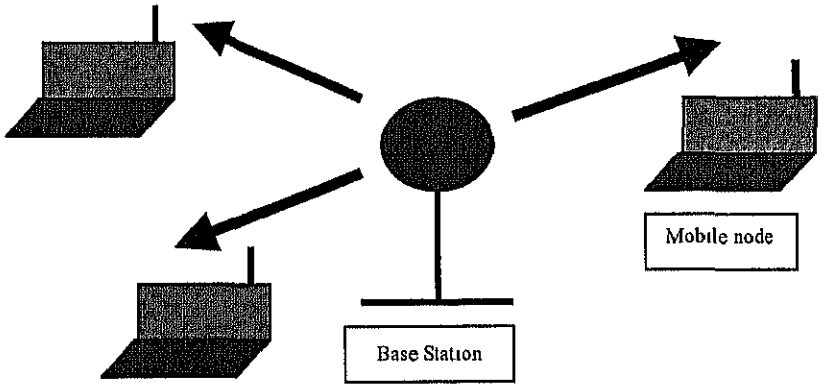



**Figure2.2** Down link traffic

**Uplink Traffic**: The uplink protocol is the core task for the MAC design of Wireless LANs. To recognize and register new mobile nodes that join the network in any time and place, a kind of random access protocol is needed. Thus uplink traffic needs a multiple access protocol to organize the transmissions from mobile nodes. In the next section **Expected features of Wireless LANs** it will be explained why multiple access is more difficult for wireless LANs than for wired LANs

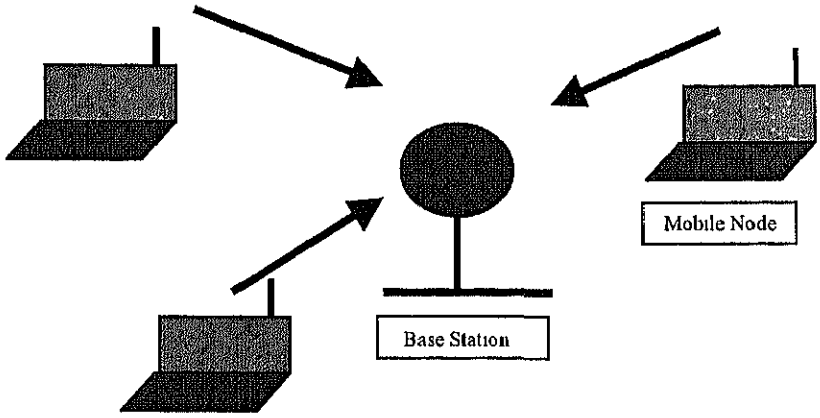



Figure 2.3 Uplink Traffic

## 2.3 Expected Features of Wireless LANs.

Multiple access is not easy in the wireless environment because of the following reasons.

### 2.3.1 Dynamic physical channel characteristics:

Wireless LANs typically operate in very strong multipath fading channels; e g the received signal may suddenly disappear or reappear. Also capture effects may occur (when two mobiles transmit at the same frequency it may be that the receiver receive very well the signal of one of them without detecting any interference). The channel statistics may change significantly within 10 to 20 ms duration or any movement of the order of one foot.

### 2.3.2 Practical implementation:

Many functions that are trivial to implement in a wired medium cannot be applied to a wireless medium. For example, carrier sensing in cable is easy but carrier sensing in radio takes at least 30 to 50 micro Seconds: Moreover, a mobile station can't detect collision while transmitting because the difference between the strengths of the signals

### 2.3.3 Mobility and network topology:

The network must maintain normal operation while its topology is changing with time

### 2.3.4 Spatial behavior and Handoff:

As explained in the previous section, infrastructure LANs are based on some access points that divide the service area of a wireless LANs into different corresponding *cells*. One of the primary reasons to adopt cellular structure is to increase the effective total bandwidth by using different frequencies in different cells This concept, known as frequency reuse, is illustrated in the following example The Fig. 2 4 shows a seven-cell structure; suppose a total of 3-B bandwidth is needed to serve users in the seven-cell area Three different frequency bands can cover this seven-cell region If frequency reuse was not employed and a single frequency band served all users in the same region, a total of 7-B bandwidth would be needed to support the same quality of service As a result of frequency reuse, the total available communication bandwidth for all users is much larger than the transmission speed Furthermore, frequency reuse not only saves the spectrum but also reduces transmission power by reducing cell size A function that allows a mobile node to communicate with the access point in a cell and then switch to the access point in another cell is called *handoff* or *handover*. The purpose of the handoff is to maintain continuous or seamless service to mobile nodes through different cell coverage Handoff is consequently a special feature to deal with the mobility issue for wireless networks

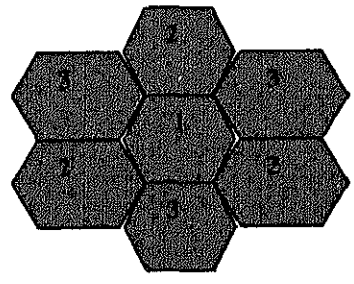



**Figure 2.4** Frequency Reuse Layout Example

Thus the expected features of a decent MAC protocol in a wireless environment should cope with all its specificities. These are some of them·

1. *Throughput*: Since spectrum is a scarce resource, throughput is definitely one of the most critical considerations in the design of a MAC protocol

2. *Delay*· Delay characteristics are important for every application, but especially for time-bounded services and multimedia applications such as voice and video.

3. *Fairness of access* Unfairness can occur because of the capture effects

4 *Battery power consumption* Efficient utilization of transmit and receive power is another important consideration for a MAC protocol

5. *Ability to support handoff between service areas.* A MAC protocol has to support a handoff function to serve nodes moving from one cell to another

6. *Establish peer-to peer connectivity* The MAC of a wireless should support ad-hoc networking

The other expected features from wireless LANs are similar to the features expected from wired LANs' ability to support multicasting, ability to support priority traffic, preservation of packet order Last, but not least, it should be possible to operate them under the Mobile-IP protocol and to support internetworking [14].

## 2.4 WIRELESS NETWORKS.

Various types of wireless networks and their specifications are as follows

| Network | Coverage | Bandwidth (in flux) | Cost | Common Use | Standards/ Protocols |
|---|---|---|---|---|---|
| Infrared (IR) | Line-of-sight Point-to-point < 6' | 9 6 Kbps to 4 Mbps | Very low | Personal Area Network (PAN) | IrDA Radio frequency |
| Bluetooth | *Omnidirectional ~30'* | *1 Mbps* | *Low* | PAN | *Bluetooth* Radio frequency 2 4GHz |
| WLAN | <100' to >300' inside, ~1 mile between buildings Shorter range with 802 11a | 11-22 Mbps w/ 802 11b ~55 Mbps with 802 11a | Low | Within buildings, between building, campus | IEEE 802 11b, 802 11a coming Radio frequency 2 4GHz |
| Wide Area Data | Regional by major city | 9 6 to 128 Kbps | Varies by application and billing plan | Major metropolitan areas, campus | Packet-switched |
| Cellular Telephony | National, spotty in rural areas | 9 6 to 14 4 Kbps (2G) 28 8 to 128 Kbps (2 5G) 300 Kbps to 2 Mbps (3G) | Varies by application and billing plan | National coverage | GSM, CDMA, TDMA, GPRS |
| Paging | National | 9 6 Kbps | Low | Two-way short text messages | CDPD |
| Satellite | Global 400 Kbps to 1,5 Mbps downlink | 256 Kbps uplink | Expensive | When broadband alternatives unavailable or max coverage | Integrated terrestrial, satellite |
| *Special-purpose* (WISPs) | *Regional by major city* | *9 6 to 128 Kbps* | Low | Single purpose, Internet/e-mail access | Needed |
| **Key** **CDMA: Code Division Multiple Access** **CDPD:** Cellular Digital Packet Data **GPRS:** General Packet Radio Service **GSM:** Global System for Mobile Communications | | | **TDMA:** Time Division Multiple Access **WISPs:** Wireless Internet/e-mail Service Providers **802.11a, 802.11b:** A family of IEEE standards for wireless LANs 802 11a defines 24 Mbps in the 5GHz band, 802.11b defines an 11-Mbps data rate in the 2 4GHz band | | |

**Table 2.1** Specifications of wireless networks

# Chapter 3

# Medium Access Control in WLANs

The wireline networks in general consist of nodes joined by point-to-point links Each such link might consist physically of a pair of twisted wire, a coaxial cable, an optical fiber, a microwave radio link etc The implicit assumption about point-to-point links, however, is that the received signal on each link depends only on the transmitted signal and noise on that link

There are many widely used communication media, such as satellite system, radio broadcast, multidrop telephone lines, and multitap bus systems, for which the received signal at one node depends on the transmitted signal at two or more other nodes. Typically such a received signal is the sum of attenuated transmitted signals from a set of other nodes, corrupted by distortion, delay, and noise Such media, called multi-access media, form the basis for LANs, MANs, satellite networks, and radio networks.

One needs an additional sublayer, often called the media access control (MAC) sublayer, between the data link control (DLC) layer and the physical layer as shown in Fig 3 1a and 3 1b. The purpose of this extra sublayer is to allocate the multi-access medium among the various nodes [2]

Conceptually, we can view multi-access communication in queuing terms Each node has queue of packets to be transmitted and multi-access channel is a common server. Ideally, the server should view all the waiting packets as one combined queue to be served by appropriate queuing discipline. Unfortunately, server does not know which nodes contain packets; similarly, nodes are unaware of packets at other nodes Thus, the interesting part of the problem is that knowledge about the state of the queue is distributed There are two extremes among the many strategies that have been developed for this generic problem. One is the "free-for-all" approach in which nodes normally send new packets immidiately, hoping for no interference from other nodes.

The interesting question here is when and how packets are retransmitted when collisions (i e interference) occur The other extreme is "perfectly scheduled" approach in which there is some order (round robin, for example) in which receive reserved intervals for channel use. The interesting question here are: (1) what determine the scheduling order (it could be dynamic), (2) how long can a reserved interval last, and (3) how are nodes informed of their turns? [2]

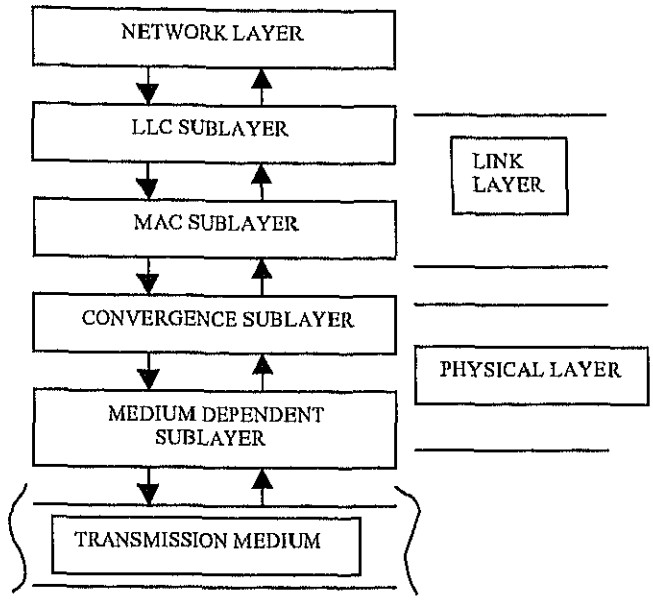



**Figure 3.1(a)** Medium Access Sublayer.

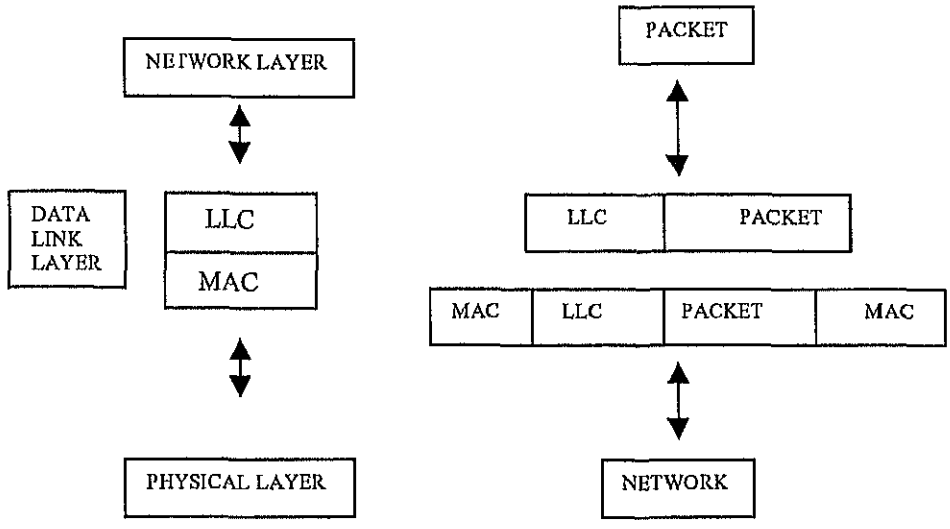



**Figure 3.1(b)** Header of MAC and LLC.

## 3.1 The Channel Allocation Problem.

In MAC layer, how to allocate a single broadcast channel among competing users is the key issue of concern. Static and dynamic channel allocation schemes are generally used

### 3.1.1 Static Channel Allocation in LANs

The basic access techniques for static channel allocation are as follows:

1. *Frequency-division-multiple-access (FDMA).* In this technique, disjoint subbands of frequency are allocated to the different users on a continuous time basis In order to reduce interference between users allocated adjacent channel bands, *guard bands* are used to act as buffer zones, as illustrated in Fig 3.2a These guard bands are necessary because of the impossibility of achieving ideal filtering FDM is the

traditional way of allocating the channel to the multiple competing users If there are N users, the bandwidth is divided into N equal sized portions, each user being assigned one portion Since each user has private frequency band, there is no interference between users. When there is only a small and fixed number of users, each of which has a heavy (buffered) load of traffic (e g , carriers switching offices), FDM is a simple allocation mechanism [3]

However, when the number of senders is large and continuously varying, or the traffic is bursty, FDM presents some problem If the spectrum is cut up into N regions, and fewer than N users are currently interested in communicating, a large piece of valuable spectrum will be wasted If more than N users want to communicate, some of them will be denied of permission, for the lack of bandwidth, even if some of the users who have been assigned a frequency band hardly ever transmit or receive anything

However, even assuming that the number of users could somehow be held constant say N, dividing the single available channel into static subchannels is inherently inefficient. The basic problem is that when some users are quiescent, their bandwidth is simply lost They are not using it, and no one else is allowed to use it either. Furthermore, in most computer systems, data traffic is extremely bursty (peak traffic to mean traffic ratios of 1000·1 are common). Consequently, most of the channels will be idle for most of the time.

The poor performance of static FDM can easily be seen from simple queuing theory calculation Let us start with mean time delay, T, for a channel of capacity C bps, with an arrival rate of $\lambda$ frames/sec, each frame having a length drawn from an exponential probability density function with mean $1/\mu$ bits/frame:

$$T = 1 / (\mu C - \lambda) \qquad (3.1)$$

Now let us divide the single channel up into N independent subchannels, each with capacity C/N bps The mean input rate on each of the subchannels will now be $\lambda/N$. Recomputing T we get

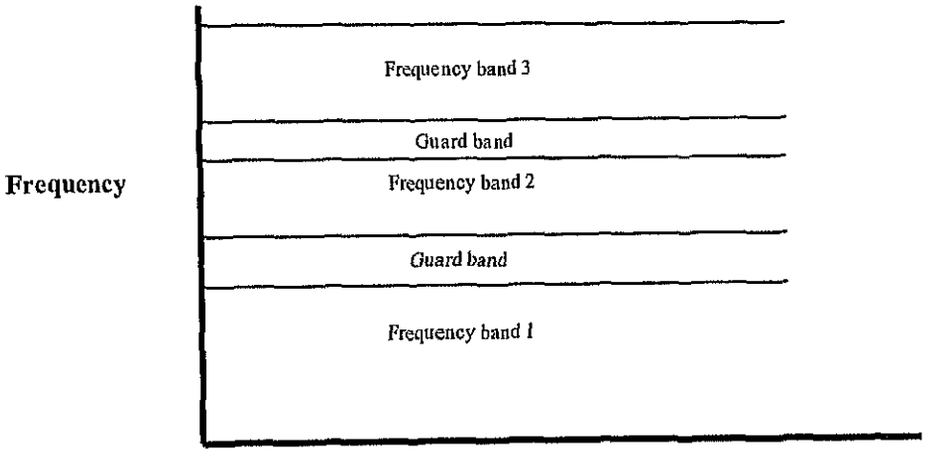
$$T_{FDM} = 1 / (\mu (C/N) - (\lambda/N)) = N / (\mu C - \lambda) = NT \qquad (3.2)$$

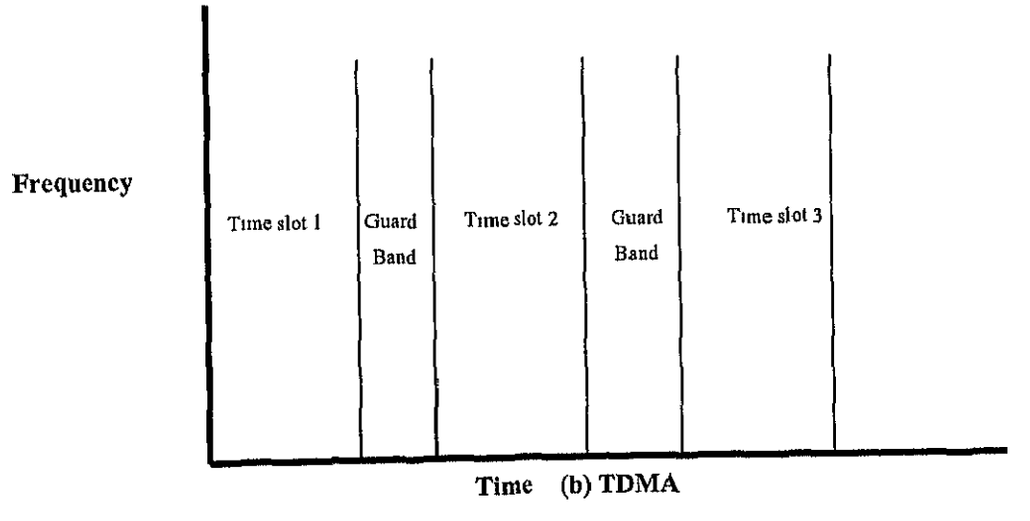
The mean delay using FDM is N times worse than if all frames were somehow magically arranged orderly in a big central queue

2 *Time-divison multiple access (TDMA).* In this technique, each user is allocated the full spectral occupancy, but only for a short duration of time called a *time slot*. As shown in Fig. 3.2b, buffer zones in the form of *guard times* are inserted between the assigned time slots. This is done to reduce interference between users by allowing for time uncertainty that arises due to system imperfections, especially in

synchronization schemes In TDM each user is statically allocated to every Nth time slot If the user does not use an allocated slot, it just lies idle.

3 *Code-divison multiple access* This technique is hybrid combination of FDMA and TDMA, which represents a specific form of generalised code-division multiple access (CDMA) Specifically, *frequency hopping* may be employed to ensure that during each successive time slot, the frequency bands assigned to the users are reordered in an essentially random manner For example during time slot 1, user 1 occupies frequency band 1, user 2 occupies frequency band 2, and user 3 occupies frequency band 3 and so on During time slot 2, user 1 hops to frequency band 3, user 2 hops to frequency band 1, user 3 hops to frequency band 2 and so on. Such an arrangement has the appearance of the users playing a game of musical chairs An important advantage of CDMA over both TDMA and FDMA is that it can provide for secure communications In the type of CDMA illustrated in Fig 3.2c, the frequency hopping mechanism can be implemented through the use of a *pseudo-noise (PN) sequence,* which is a cyclic code with noise like characteristics. [4]



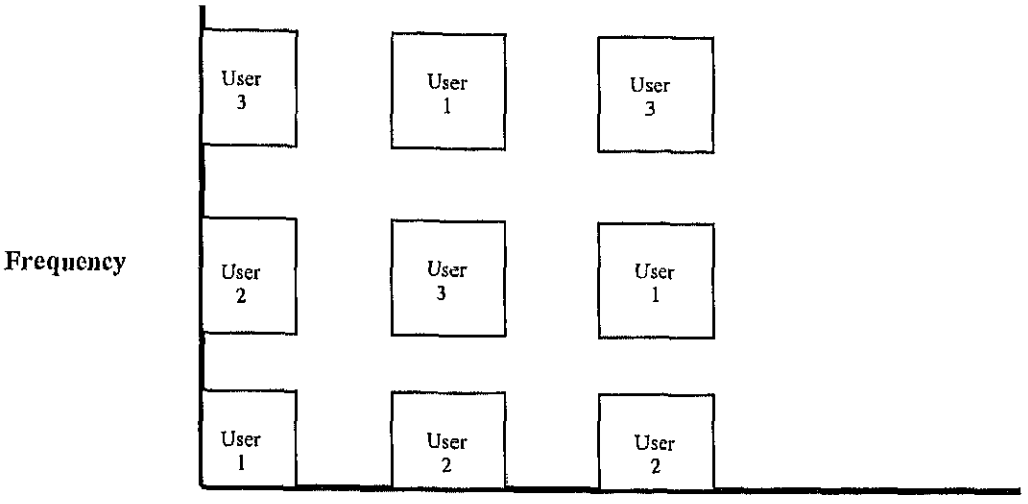



**Figure 3.2** Multiple-access techniques

### 3.1.2 Dynamic Channel Allocation in LANs

Before we get into the first of the many channel allocation methods to be discussed in this chapter, it is worthwhile carefully formulating the allocation problem. Underlying, all the work done in this area are five key assumptions described below

1. *Station model.* The model consists of N independent stations, each with a program or user that generates frames for transmission. The probability of a frame being generated due to an arrival in duration $\Delta t$ is $\lambda\Delta t$, where $\lambda$ is a constant (arrival rate of new frames) Once a frame has been generated, the station is blocked and does nothing until the frame has been successfully transmitted
2. *Single Channel Assumption* A single channel is available for all communication. All stations can transmit on it and all can receive from it. As far as the hardware is concerned, all stations are equivalent, although protocol software may assign priorities to them
3. *Collision Assumption.* If two frames are transmitted simultaneously, they overlap in time then resulting signal is garbled This event is called a **collision.** All station can detect collisions. A collided frame must be transmitted again later. There are no errors other than those generated by collisions.

4a *Continuous Time.* Frame transmission can begin at any instant. There is no master clock dividing time into discrete intervals.

4b *Slotted Time* Time is divided into discrete intervals (slots) Frame transmissions always begin at the start of a slot A slot may contain0, 1, or more frames, corresponding to an idle slot, a successful transmission, or a collision, respectively

5a Carrier *Sense* Stations can tell if the channel is in use before trying to use it If the channel is sensed as busy, no station will attempt to use it until it goes idle

5b No *Carrier Sense.* Stations can not sense the channel before trying to use it They just go ahead and transmit Only later can they determine whether or not the transmission was successful LANs generally have carrier sense, but satellite networks do not (due to long propagation delays) Stations on carrier[1] sense networks can terminate their transmission prematurely if they discover that it is colliding with other transmission Many algorithms for allocating multiple access channels are known. In the following sections we will discuss some of them

## 3.2 ALOHA.

Norman Abramson and his colleagues at the University of Hawaii devised a new and elegant method to solve the allocation problem, called ALOHA We will discuss two versions of ALOHA here. pure and slotted. They differ with respect to whether or not time is divided up into discrete slots into which all frames must fit Pure ALOHA does not require global time synchronization; slotted ALOHA does.

### 3.2.1 Pure ALOHA.

The basic idea of an ALOHA system is simple. let users transmit whenever they have data to be sent. There will be collisions, and the colliding frames will be destroyed. However, due to feedback property of broadcasting, a sender can always find out whether or not its frame was destroyed by listening to the channel, the same way other users do With a LAN the feedback is immediate Systems in which multiple users share a common channel in away that can lead to conflicts are widely known as contention systems

A sketch of frame generation in an ALOHA system is given in Fig. 3.3. We have made the frames all of the same length because the throughput of ALOHA systems is maximized by having a uniform frame size rather than allowing variable length frames.

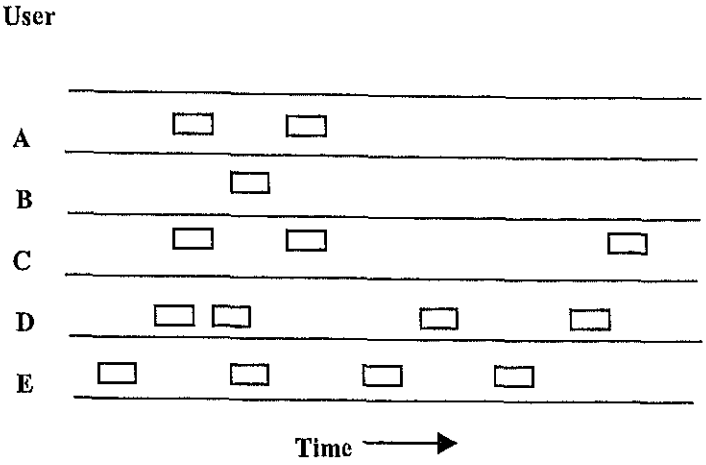



**Figure 3.3** In pure ALOHA, frames are transmitted at completely arbitrary times

Whenever two frames try to occupy the channel at the same time, there will be a collision and both will be garbled. If the first bit of a new frame overlaps with just the last bit of a frame almost finished, both frames will be totally destroyed, and both will have to retransmitted later The checksum cannot distinguish between a total loss and a near miss Let the frame time denote the standard, fixed length frame (i e , the frame length divide by the bit rate). We assume that the infinite population of the users generates new frames according to the Poisson distribution with mean S frames per frame time. If S > 1, the user community is generating frames at a higher rate than the channel can handle, and nearly every frame will suffer a collision. For reasonable throughput S would be (0,1).

In addition to the new frames, the stations also generate retransmissions of frames that previously suffered collisions Let us further assume that the probability of $k$ transmission attempts per frame time, old and new combined, is also Poisson with mean G per frame time Clearly $G \geq S$. At low load, (i.e., $s \approx 0$), there will be few collisions, hence few retransmissions, so $G \approx S$. At high load there will be many collisions, so $G > S$. Under all loads, the throughput is just the offered load, G, times the probability of a transmission being successful i e , $S = GP_0$, where $P_0$ is the probability that a frame does not suffer a collision.

A frame will not suffer a collision if no other frames are sent within one frame time of its start, as shown in Fig. 3.4. Let t be the time required to send a frame If any other user has generated a frame between time $t_0$ and $t_0 + t$, the end of that frame will collide with beginning of the shaded one. Similarly, any other frame starting between $t_0 + t$ and $t_0 + 2t$ will bump into the end of the shaded frame

---

[1] The word "carrier" in this sense refers to an electrical signal on the cable

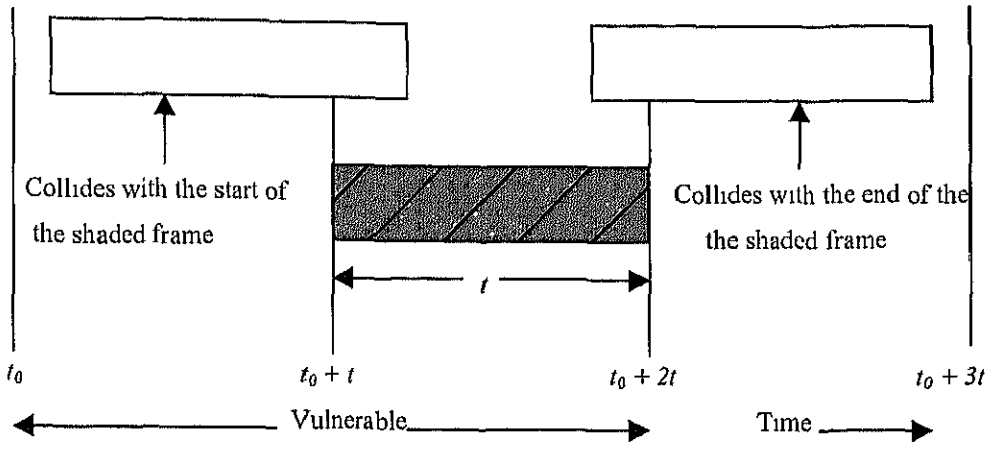



**Figure 3.4** Vulnerable period for the shaded frame.

The probability that $k$ frames are generated during a given frame time is given by the Poisson distribution

$$\Pr[k] = G^k e^{-G} / k! \tag{3.3}$$

So the probability of zero frames is $e^{-G}$. In an interval two-frame time long, the mean number of frames generated is 2G. The probability of no other traffic being initiated during the entire vulnerable period is thus given by $P_0 = e^{-2G}$ Using $S = GP_0$, we get

$$S = G e^{-2G} \tag{3.4}$$

The relation between the offered traffic and the throughput is shown in Fig 3 5. The maximum throughput occurs at G = 0.5, with S = 1/2e, which is about 18 percent

### 3.2.2 Slotted ALOHA.

In order to double the capacity of ALOHA time is divided up into discrete intervals, each interval corresponding to one frame, but due to discrete intervals the problem of timing synchronization arises. One way to achieve synchronization would be to have special station emit a pip at the start of each interval In slotted ALOHA, in contrast to pure ALOHA a computer is not permitted to send whenever a carriage return is typed. Instead, it is required to wait for a next time slot. Since the vulnerable period is halved, the probability of no other traffic during the same slot is $e^{-G}$ which leads to

$$S = G e^{-G} \tag{3.5}$$

As from Fig. 3 5 slotted ALOHA peaks at G = 1, with a throughput of S = 1/e, which is twice that of pure ALOHA

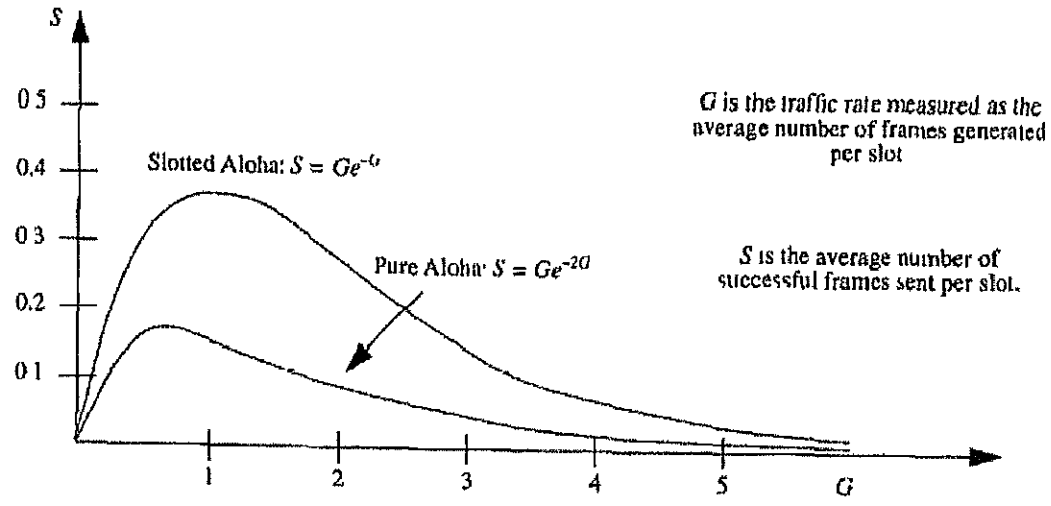



**Figure 3.5** Throughput (S) versus Offered Load (G) Plot for ALOHA and S-ALOHA [5]

## 3.3 CSMA Transmission Protocols.

The various protocols considered below differ by the action (pertaining to packet transmission) that a terminal takes after sensing[2] the channel However, in all cases, when a terminal learns that its transmission was unsuccessful, it reschedules the transmission of the packet according to a randomly distributed retransmission delay At this new point in time, the transmitter senses the channel and repeats the algorithm dictated by the protocol. At any instant a terminal is called the *ready terminal* if it has a packet ready for transmission at this instant (either a new packet just generated or a previously conflicted packet rescheduled for transmission at this instant) [5]

A terminal may, at any one time, either be transmitting or receiving (but not both simultaneously). However, the delay incurred to switch from one mode to the other is negligible Furthermore, time required to detect the carrier due to packet transmission is negligible (i.e. a zero detection time is assumed).[3] All packets are of constant length and are transmitted over an assumed noiseless channel (i e. the errors in the packet reception caused by random noise are not considered to be a serious problem and are neglected in comparison with errors caused by overlap interference). The system assumes noncapture (i.e. the overlap of any fraction of two packets results in destructive interference and both packets must be

[2] Each terminal has the capability of sensing carrier on the channel

[3] The detection time is considered negligible for relatively wideband channels (100 kHz)

retransmitted) We further simplify the problem by assuming the propagation delay (small compared to packet transmission time) to be identical[4] for all source destination pairs

We first consider the nonpersistant *CSMA* The idea here is to limit the interference among packets by always rescheduling a packet, which finds the channel busy upon arrival More, precisely a ready terminal senses the channel and operates as follows.

1 If the channel is sensed idle, it transmits the packet.

2. If the channel is sensed busy, then the terminal schedules the retransmission of the packet to some later time according to the retransmission delay distribution At this new point in time, it senses the channel and repeats the algorithm described

A slotted version of the nonpersistent CSMA can be considered in which the time axis is slotted and the slot size is $\tau$ seconds (the propagation delay) All terminals are synchronized and are forced to start transmission only at the beginning of a slot When a packet arrival's occurs during a slot, the terminal senses the channel at the beginning of the next slot and operates according to the protocol described above.

We next consider the *p-persistent CSMA* protocol However, before treating the general case (arbitrary p), we introduce the special case of $p = 1$

The *1-persistent CSMA* protocol is devised in order to achieve acceptable throughput by never letting the channel go idle if some ready terminal is available More, precisely a ready terminal senses the channel and operates as follows

1. If the channel is sensed idle, it transmits the packet with probability one

2. If the channel is sensed busy, it waits until the channel goes idle (i.e., persistent on transmitting) and only then transmits the packet (with probability one-hence, the name of 1-persistent).

A slotted version of 1-persistent CSMA can also be considered by slotting the time axis and synchronizing the transmission of packets in much the same way as the previous protocol.

The above 1-persistent and nonpersistent protocols differ by the probability (zero or one) of not rescheduling the packet which upon arrival finds the channel busy. In the case of a 1-persistent CSMA, we note that whenever two or more terminals become ready during a transmission period (TP), they wait for the channel to become idle (at the end of that transmission) and then they all transmit with probability one. A conflict will also occur with probability one! The idea of randomizing the starting time of

[4] By considering this constant propagation delay equal to largest possible, one gets lower bounds on the performance.

transmission of packets accumulating at the start of a TP (Transmission Period) suggests itself for the interference reduction and throughput improvement The scheme consists of including an additional parameter $p$, the probability that a ready packet persists (1-$p$ being the probability of delaying the transmission by $\tau$ seconds) The parameter $p$ to be chosen as to reduce the level of interference while keeping the idle periods between any two consecutive nonoverlapped transmission as small as possible This gives rise to *p-persistent CSMA*, which is generalization of the 1-persistent CSMA

More precisely the protocol consists of the following the time axis is finely slotted where the (mini) slot size is $\tau$ seconds For simplicity of analysis, we consider the system to be synchronized such that all packets begin their transmission at the beginning of a (mini) slot

Consider a ready terminal. If the channel is sensed idle, then with probability $p$, the terminal transmits the packet; or with probability $1-p$, the terminal delays the transmission of packet by $\tau$ seconds (i.e., one time slot). If at this new point in time, the channel is still detected idle, the same process is repeated. Otherwise, some packet must have started transmission, and our terminal schedules the retransmission of packet according to the retransmission delay distribution (i e., acts as if it had conflicted and learned about the conflict)

If the ready terminal senses the channel busy, it waits until it becomes idle (at the end of current transmission) and then operates as above The comparisons of various protocols i.e , ALOHA and CSMA are shown in Fig. 3.6

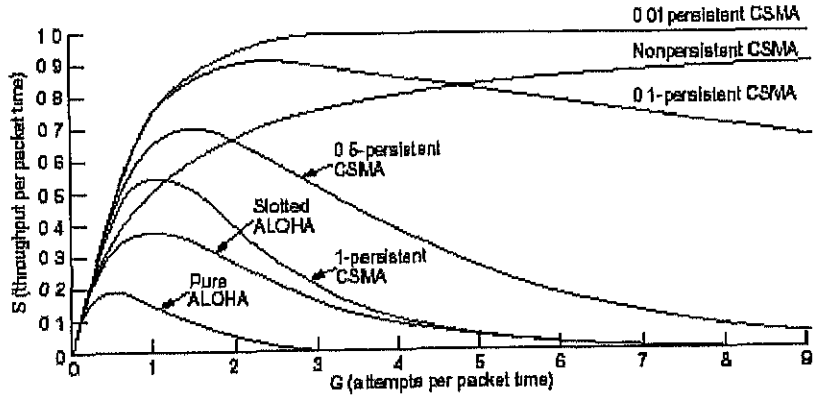



**Figure 3.6** Comparison of CSMA and ALOHA protocols [5].

# Chapter 4

# MAC Protocols in Wireless LANs

In recent years, a wide variety of mobile computing devices have emerged, including portables, palmtops, and personal digital assistants Providing adequate network connectivity for these devices will require a new generation of wireless LAN technology Many medium access control (MAC) protocols for wireless networks proposed and implemented to date are based on collision avoidance handshakes between sender and receiver In this chapter, we will study medium access protocols for a single channel wireless LAN We start with MACA [6], MACAW [7], IEEE 802 11 [8], FAMA [9], MACA-BI [10] and RIMA [11]. MAC protocols in WLANs can be categorised into two types: sender initiated and receiver initiated The sender-initiated protocols in above are MACA, MACAW, IEEE 802 11, FAMA and the receiver-initiated protocols are MACA-BI and RIMA.

## 4.1 Sender initiated MAC protocols.

Various sender initiated MAC protocols are as follows.

### 4.1.1 MACA and MACAW.

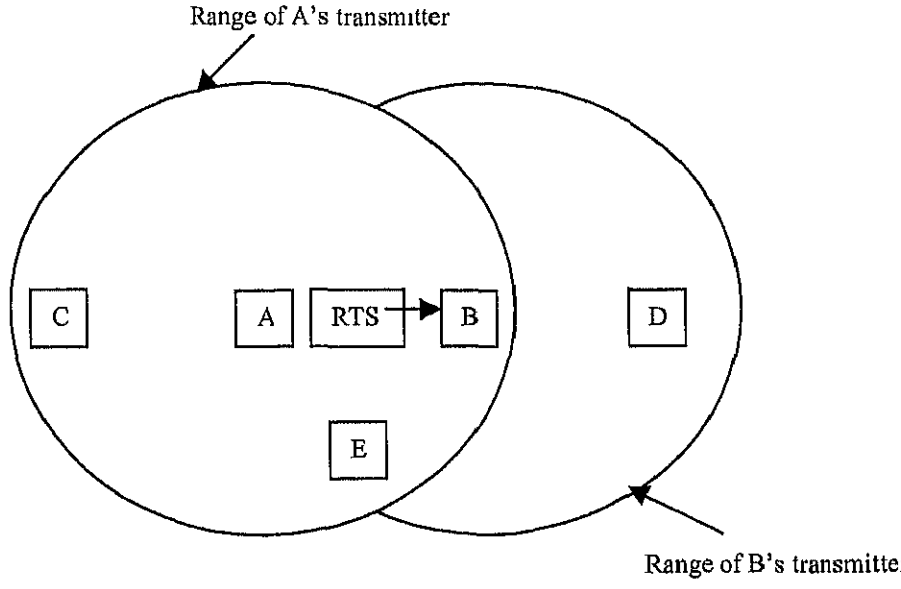
An early protocol designed for wireless LANs is MACA (Multiple Access with Collision Avoidance) (Karn, 1990) It was used as a basis of IEEE 802.11 wireless LAN standard The basic idea behind it is for the sender to stimulate the receiver into outputting a short frame, so stations nearby can detect this transmission and avoid transmitting themselves for the duration of the upcoming (large) data frame. MACA is illustrated in Fig 4.1.

Let us consider how 'A' sends a frame to 'B' 'A' starts by sending an RTS (Request To Send) frame to 'B', as shown in Fig 4 1 (a). This short frame contains the length of the data frame that will eventually follow Then 'B' replies with a CTS (Clear To Send) frame, as shown in Fig 4 1 (b) The CTS frame contains the data length (copied from the RTS frame). Upon receipt of the CTS frame, 'A' begins transmission.

Now let us see how stations overhearing either of these frames react Any station hearing the RTS is clearly close to 'A' and must remain silent long enough for the CTS to be transmitted back to 'A' without

conflict Any station hearing the CTS is clearly close to 'B' and must remain silent during the upcoming data transmission, whose length it can tell by examining the CTS frame

In Fig 4.1, 'C' is within range of 'A' but not within range of 'B' Therefore it hears the RTS from 'A' but not the CTS from 'B'. As long as it does not interfere with the CTS, it is free to transmit while the data frame is being sent In contrast 'D' is within range of 'B' but not 'A'. It does not hear the RTS but does hear the CTS Hearing the CTS tips it off that it is close to a station that is about to receive a frame, so it defers from sending anything until that frame is expected to be finished Station 'E' hears both control messages, and like 'D', must be silent until the data frame is complete




**(a)**

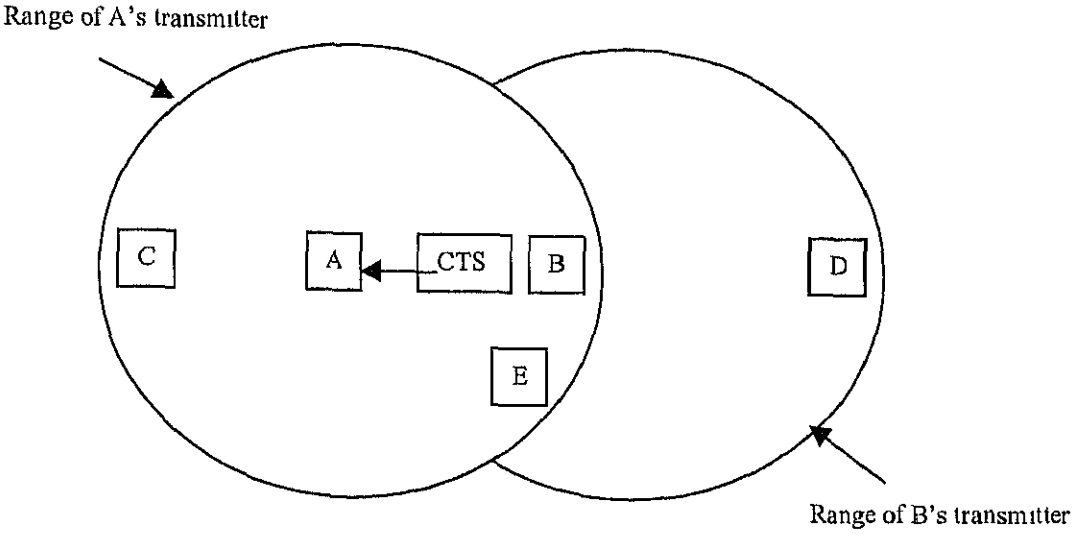



**Figure 4.1.** The MACA protocol. (a) A sending a RTS to B (b) B responding with a CTS to A

Despite these precautions, collisions can still occur. For example, 'B' and 'C' could both send RTS frames to 'A' at the same time. These will collide and be lost. In the event of a collision, an unsuccessful transmitter (i e., one that does not hear CTS within the expected time interval) waits a random amount of time and tries again later. The algorithm used is binary exponential backoff.

Based on simulation study of MACA, Bharghavan et al fine tuned MACA to improve its performance and renamed their new protocol MACAW. To start with, they noticed that without data link layer acknowledgements, lost frames were not retransmitted until the transport layer noticed their absence, much later They solved this problem by introducing an ACK frame after each successful data frame. They also observed that CSMA has some utility—namely to keep a station from transmitting an RTS at the same time another nearby station is also doing so to the same destination, so the carrier sensing was added In addition they decided to run backoff algorithm separately for each data stream (source-destination pair), rather than for each station, this change improves the fairness of protocol. Finally, they added a mechanism for stations to exchange information about congestion, and a way to make the backoff algorithm react less violently to temporary problems, to improve system performance.

## 4.1.2 IEEE 802.11.

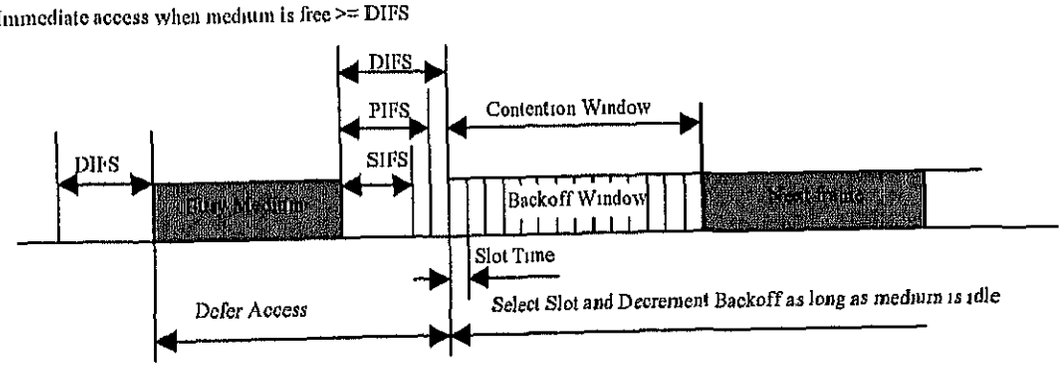
The IEEE 802 11 MAC protocol is based on a *Carrier Sense Multiple Access with Collision Avoidance* (CSMA/CA) protocol The time to sense the carrier is defined by the *Interframe space* (IFS) The collision avoidance is done by a random backoff procedure [12].

### 4.1.2.1 Interframe Space.

The time between two frames is called an *interframe space* (IFS) In order to determine whether the medium is free, a station has to use a carrier sense function for specified IFS The standard specifies four different IFSs, which represent three different priority levels for the channel-access The shorter the IFS, higher the priority. The IFSs are specified as time gaps on the medium and are independent of the channel data rate. Owing to the different characteristics of the different PHY specifications, the IFS time duration is specific for each PHY. Some relations between the IFSs are shown in Fig. 4 2 The IFS are listed in order, from shortest to longest [13].

**Short IFS (SIFS).** The SIFS is used for the immediate acknowledgement (ACK frame) of a data frame, the answer (*Clear To Send (CTS) frame*) to a *Request To Send* (RTS) frame, a subsequent MPDU (MAC Protocol Data Unit) of a fragmented MSDU (MAC Service Data Units), response to any polling by the PCF (Point Co-ordination Function), and any frames of the AP (Access Point) during the *Contention – Free Period* (CFP)

**Point Co-ordination Function IFS (PIFS).** The PIFS are used by the stations operating under the PCF (Point Co-ordination Function) to gain access to the medium at the start of the CFP.




SIFS = Short Interframe Space, DIFS = Distributed Co-ordination Function Interframe Space, PIFS = Point Co-ordination Function Interframe Space

**Figure 4.2** Inteframe space relationship.

**Distributed Co-ordination Function IFS (DIFS).** The DIFS is used by the stations operating under the DCF (Distributed Co-ordination Function) to gain access to the medium to transmit data or management frames.

**Extended IFS (EIFS).** The EIFS is used by the DCF whenever the PHY indicates that a frame transmission did not result in a correct *frame check sequence* (FCS) The EIFS allows another station to acknowledge what was, to this station, an incorrectly received frame.

**4.1.2.2 Distributed Co-ordination Function.**

According to the DCF (in Fig 4 3), a station must sense the medium before initiating the transmission of a packet If the medium is sensed as being idle for a time interval greater than a DIFS then the station transmits the packet Otherwise, the transmission is deferred and the backoff is started Specifically, the station computes a random number uniformly distributed between zero and a maximum called *Contention Window* (CW). The random number is multiplied by the *slot time*, resulting in the backoff interval used to set the backoff timer This timer is decremented only when the medium is idle, whereas it is frozen when another station is transmitting. Each time the medium become idle, the station waits for a DIFS and then periodically decrements the backoff timer.

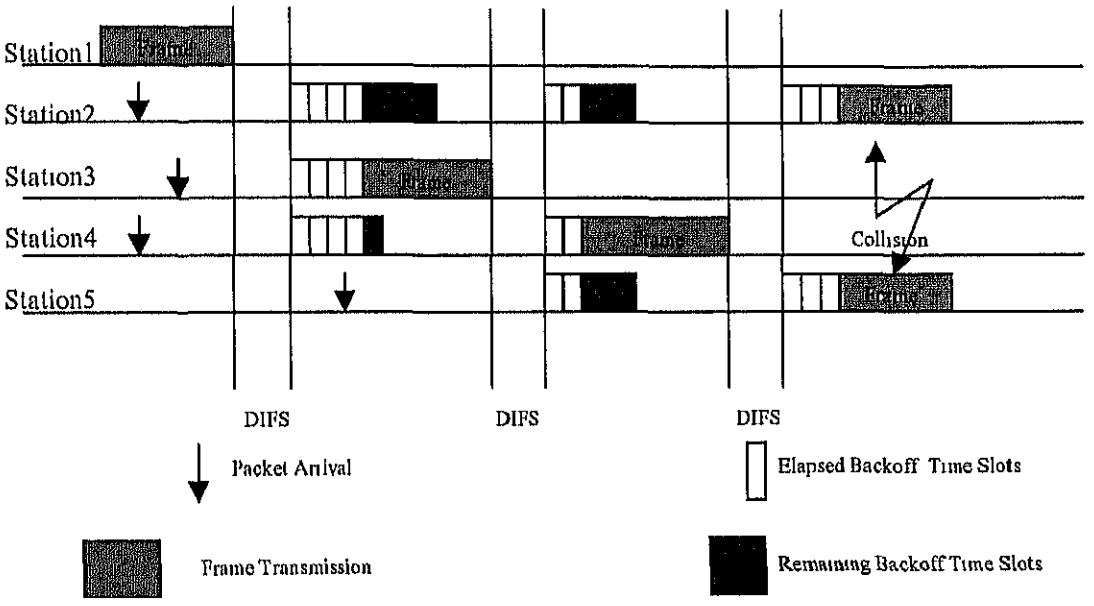



**Figure 4.3** Basic Access Mechanism.

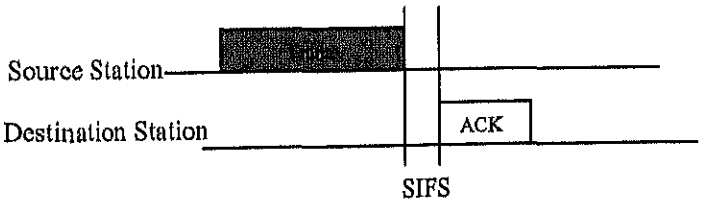



**Figure 4.4** Acknowledgement mechanism.

As soon as backoff timer expires, the station is authorised to access the medium If two or more stations start transmission simultaneously, a collision occurs Unlike, wired networks, in wireless environment collision detection is not possible because of half-duplex radios Hence as shown in Fig 4 4, a positive acknowledgement is used to notify the sending station that the transmitted frame has been successfully received The transmission of the acknowledgement is initiated at a time interval equal to the SIFS after the end of the reception of the previous frame

If the acknowledgement is not received in the specified time interval, the station assumes that the transmitted frame was not successfully received, and hence schedules a retransmission and enters the backoff process again. However, to reduce the probability of collisions, after each unsuccessful transmission attempts the *Contention Window* is doubled until a predefined maximum ($CW_{max}$) is reached After, a successful transmission, the *Contention Window* is reset to $CW_{min}$ After each frame transmission, a station must execute a new backoff process. Therefore, at least one backoff is in between two transmissions of the same station.

In radio systems based on medium sensing, a phenomenon known as the *hidden-station problem* may occur. This problem arises when a station is able to successfully receive frames from two different stations but the two stations can not receive signals from each other In this case a station may sense the medium as being idle even if the other one is transmitting This results in a collision at the receiving station.

To deal with the hidden-terminal problem, the IEEE 802 11 MAC protocol includes a mechanism based on the exchange of two short control frames (as shown in Fig 4 5) a *Request-To-Send* (RTS) frame that is sent by a potential transmitter to the receiver and a *Clear-To-Send* (CTS) frame that is sent by the receiver in response to the received RTS frame. If the CTS frame is not received within the predefined time interval, the sender node, executing the backoff algorithm described above retransmits the RTS frame. After a successful exchange of the RTS and CTS frames, the transmitter can send the data frame after waiting for a SIFS. The implementation of RTS packet is optional, whereas all stations must be able to answer to a RTS with the belonging CTS.

The RTS and CTS frames include a duration field that specifies the time interval necessary to completely transmit the data frame and the related acknowledgement This information is used by stations that can hear either the transmitter or the receiver to update their Net Allocation Vector (NAV), a timer that, unlike the backoff timer, is continuously decremented irrespective of the status of the medium.

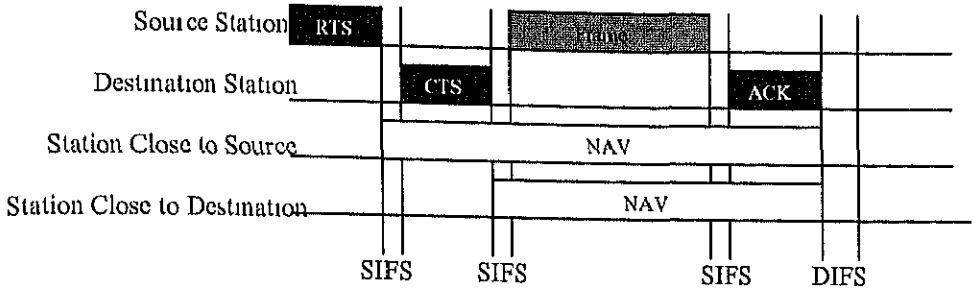



**Figure 4.5** RTS/CTS mechanism.

Since stations that can hear either the transmitter or the receiver refrain from transmitting until their NAV has expired, the probability of a collision occurring because of a hidden station is reduced Ofcourse, the drawback of using RTS/CTS mechanism is an increased overhead, which may be significant for short data frame

Furthermore, the RTS/CTS mechanism can be regarded as a way to improve the MAC protocol performance In fact, when the mechanism is enabled, collisions can obviously occur only during the transmission of the RTS frame Since the RTS frame is usually much shorter than the data frame, the waste of bandwidth and time due to the collision is reduced.

In both cases the effectiveness of the RTS/CTS mechanism depends upon the length of the data frame to be protected. Consequently, the RTS/CTS mechanism relies on a threshold, the *RTS threshold*: the mechanism is enabled for data frame sizes over the threshold and disabled for data frame sizes under the threshold The RTS/CTS is useful also while operating overlapping BSS (Basic Service Set) or IBSS (Independent Basic Service Set).

#### 4.1.2.3 Point Co-ordination Function.

In order to support time-bounded services, the IEEE 802.11 standard defines the *Point Co-ordination Function* (PCF) to permit a *Point Co-ordinator* (PC) to have priority access to the medium Usually an AP (Access Point) in an infrastructure based network acts as PC.

Although, PCF is optional, all stations are able to obey the medium access rules of the PCF, because it is based on the DCF. Stations that are able to respond to polls by the PC are called *Contention-Free-Pollable* (CF-Pollable). Only these stations are able to transmit frames according to the PCF (besides the AP)

The PCF controls the frame transfers during the so-called *Contention-Free Period* (CFP), which alternates with the *Contention Period* (CP) under the control of the DCF The CFP is periodically repeated in time at the Contention-Free Repetition Rate (CFP Rate) and starts with the transmission of a beacon. The beacon contains the maximum duration of the CFP (CFPMaxDuration), and all stations in the BSS except the PC set then NAV to CFPMaxDuration, thus guaranteeing the control of the PCF for this amount of time Fig 4.6 shows this relation.

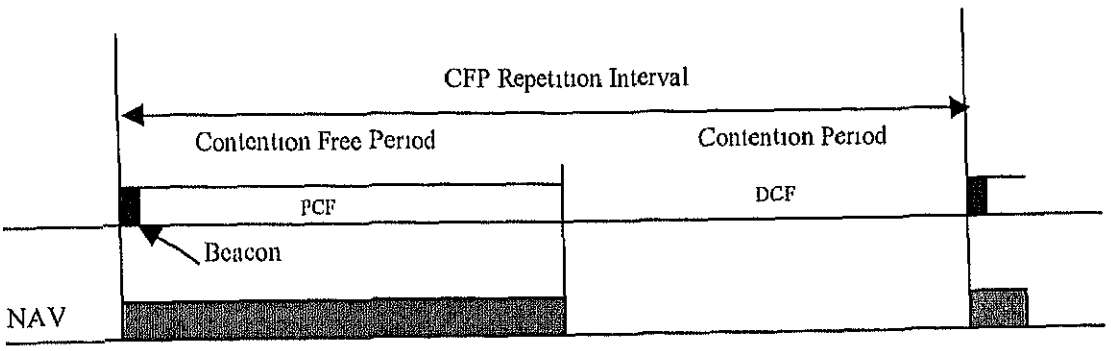



**Figure 4.6** Relationship between CFP and CP

The PC gains control at the nominal beginning of the CFP by waiting PIFS after the medium is sensed idle instead of DIFS. It maintains control for the entire CFP by waiting a shorter time than stations using the DCF access procedure The PC maintains a *Polling List*, which consists of *Association Identifier* (AID) of the stations requesting polling. A *CF-Pollable* station may request to be added to the polling list during *Association* or *Reassociation*.

Special Data Subtypes are defined for the use during CFP in order to enable "piggybacking" of polls and acknowledgements (see Table 4.1). A *CF-Poll* is used by the PC to poll a station for the transmission of a data frame. *CF-Ack* is the acknowledgement to a successfully received frame under the PCF, either by a station or the PC. The *Null Function* is used to indicate that no data has to be transmitted If all stations on the polling list have been polled and no more data has to be transmitted by the PC during one CFP, the PC may prematurely stop the current CFP by sending a *CF-End*. On receiving a *CF-End*, all stations reset their NAV

Fig. 4.7 shows an example of a sequence of frame transmissions during a CFP. Usually the gap between the two transmissions under the PCF is SIFS unless a station does not respond to a *CF-Poll*. In the later case the PC regains control of the medium after a PIFS

**Table 4.1** 802 11 MAC Data Subtypes under the PCF.

| Station | Data Subtypes |
|---|---|
| PC | Data+CF-Poll, Data+CF-Ack+CF-Poll, CF-Poll, CF-Ack+CF-Poll, CF-End+CF-Ack |
| PC and CF-Pollable | Data, Data+CF-Ack, CF-Ack, Null Function |

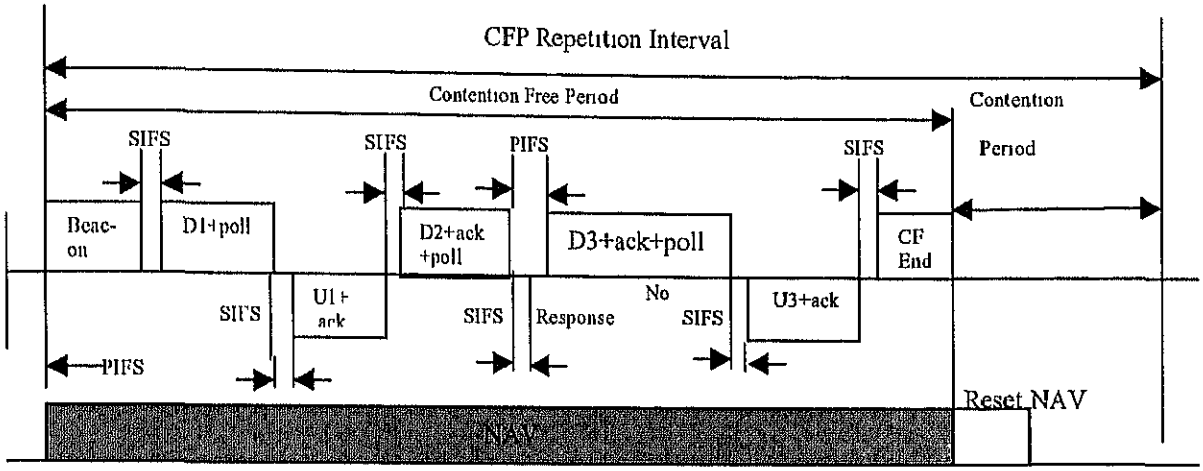



**Figure 4.7** Example of PCF frame transfer.

## 4.2 Receiver Initiated MAC Protocols.

This section introduces new MAC protocols based on receiver initiated collision avoidance and relate them to the taxonomy of polling disciplines. To our knowledge, these protocols are the first based on receiver-initiated collision avoidance that eliminate the collisions of data packets with any other control or data packets in the presence of hidden terminals For simplicity, we describe the new MAC protocols without the use of acknowledgements (ACKs); in practice, ACKs will be used. However, it should be clear that, because the protocols support correct collision avoidance, an acknowledgement to each data packet can be sent collision-free by the receiver immediately after it processes the data packet The only caveat is that the time that a node must back off to let data flow without collisions must include the time needed for the sender to receive the acknowledgement in the clear

### 4.2.1 MACA-BI.

The original MACA-BI [10] protocol uses a ready-to-receive packet (RTR) to invite a node to send a data packet. A node is allowed to send a data packet only if it has previously received an RTR, whereas a node that receives an RTR that is destined to a different node has to back off long enough for a packet to be sent in the clear. According to the description of MACA-BI, a polled node can send a data packet intended to the polling node or any other neighbour. In a fully-connected network, whether the data packet is sent to the polling node or not is not important, because all the nodes must back off after receiving an RTR in the clear. However, this is not the case in a network with hidden terminals. By means of two simple examples, we can show that MACA-BI does not prevent data packets sent to a given receiver from colliding with other data packets sent concurrently in the neighbourhood of the receiver. The first example illustrates the fact, that in order to avoid the transmission of data packets that the intended receiver cannot hear because of other colliding data packets, a polled node should send data packets only to the polling node. The second example illustrates the possibility that collisions of data packets at a receiver may occur because the receiver sent an RTR at approximately the same time when data meant for another receiver starts arriving. In Fig. 4.8, nodes 'A' and 'D' send RTRs to nodes 'B' and 'E' at time $t_0$, respectively. This prompts the polled nodes to send data packets at time $t_1$; the problem in this example occurs when at least one of the polled nodes sends a data packet addressed to 'C', which cannot hear either packet.

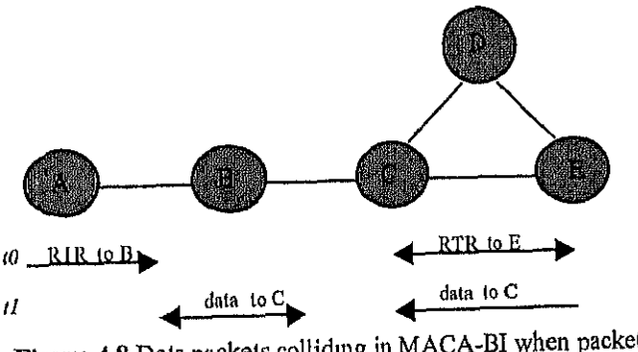



**Figure 4.8** Data packets colliding in MACA-BI when packet is not sent to polling node

In the example shown in Fig. 4.9, node 'A' sends an RTR to 'B' at time $t_0$. This RTR makes node 'B' start sending data to node 'A' at time $t_1$ which in order to provide good throughput must be larger than $\gamma$ seconds, where $\gamma$ is the length of an RTR. At time $t_2$ node 'C' starts sending an RTR to node 'D'. Because of carrier sensing, $t_2$ must be within $\tau$ seconds (maximum propagation delay) of t1. In this example, after

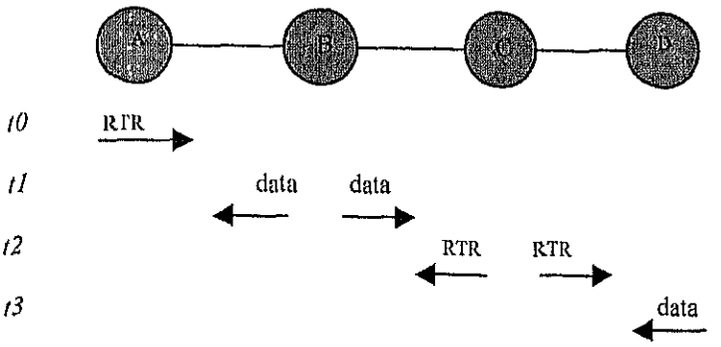



**Figure 4.9** Data packets colliding in MACA-BI due to RTR not being heard

receiving node C's RTR, node 'D' replies with data that must start arriving at node 'C' at time $t_3$ Because the maximum propagation delay is $\tau$, it must be true that $t_3 \leq t_2+\gamma+2\tau \leq t_1+\gamma+3\tau$. Hence, if data packets last longer than $\gamma+3\tau$ seconds, the data packets from 'B' and 'D' collide at node 'C' In practice, data packets must be much longer than RTRs to provide good throughput, and it thus follows that MACA-BI cannot prevent all data packets from experiencing collisions

### 4.2.2 RIMA-SP (Simple Polling).

To make the RTR-data handshake in MACA-BI collision free, the following two minor modifications are required:

1. The polled node should transmit data packets only if they are addressed to the polling node

2 A new control signal is also required, which we call No-Transmission-Request (NTR), and an additional collision-avoidance waiting period of $\xi$ seconds is required at a pollednode prior to answering an RTR During that period, if any channel activity is heard, the receiver (polling node) that originated an RTR sends an NTR telling the polled node not to send any data. Otherwise, if nothing happens during the waiting period, the polled sender transmits its data, if it has any to send to the polling node.

We call the protocol resulting from modifying MACA-BI with the above two rules RIMA-SP (receiver initiated multiple access with simple polling). Fig. 4.10 illustrates the operation of RIMA-SP. The RIMA-SP provides correct collision avoidance when $\xi = \tau$ [11]. In RIMA-SP, every node initialises itself in the START state, in which the node waits twice the maximum channel propagation delay, plus the hardware transmit-to-receive transition time ($\varepsilon$), before sending anything over the channel This enables the node to find out if there are any ongoing transmissions. After a node is properly initialised, it makes transition to the PASSIVE state. In all the states, before transmitting anything to the channel, a node must listen to the channel for a period of time that is sufficient for the node to start receiving packets in transit

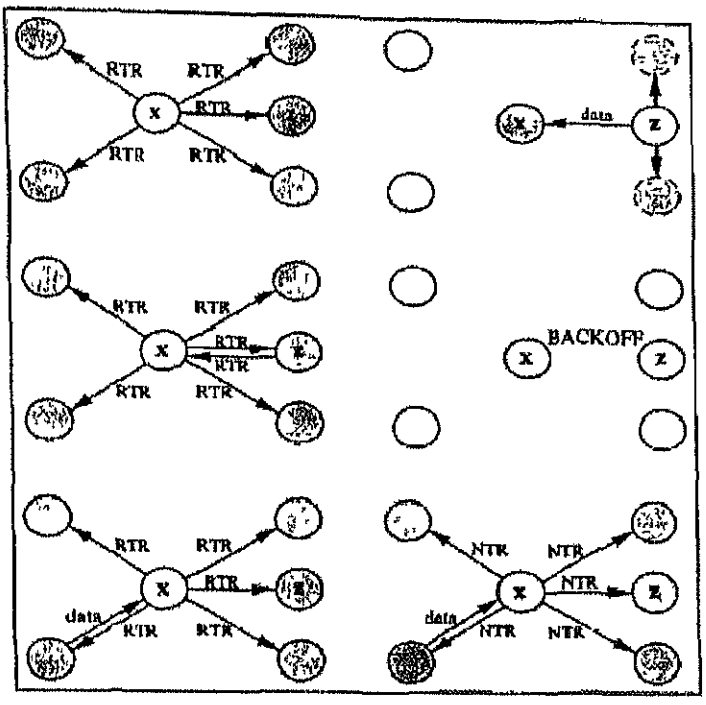



*Figure 4.10 RIMA-SP illustrated [11]*

If a node 'x' is in the PASSIVE state and senses carrier, it transitions to the REMOTE state to defer to ongoing transmissions. A node in REMOTE State must allow enough time for a complete successful handshake to take place, before attempting to transition from remote state.

Any node in PASSIVE State that detects noise in the channel must transition to the BACKOFF State. If node 'x' is in PASSIVE state and obtains an outgoing packet to send to neighbour 'z', it transitions to the RTR state. In the RTR State, node 'x' uses non-persistent carrier sensing to transmit an RTR. If node 'x' detects carrier when it attempts to send the RTR, it transitions to the BACKOFF state, which makes the node back off immediately for a sufficient amount of time to allow a complete handshake between a sender-receiver pair to occur; otherwise, 'x' sends its RTR

If node 'z' receives the RTR correctly and has data for 'x', it waits for $\xi$ seconds If during the waiting period there is no activity in the channel, node 'z' transitions to the XMIT state, where it transmits a data packet to 'x'; otherwise, node 'z' assumes that there was a collision and transitions to the BACKOFF state to allow floor acquisition by some other node. After sending its RTR, node 'x' senses the channel. If it detects carrier immediately after sending its RTR, node 'x' assumes that a collision or a successful data transfer to a hidden node is taking place. Accordingly, it sends a No transmission Request (NTR) to 'z' to stop 'z' from sending data that would only collide at 'x'. When multiple RTRs are transmitted within a one-way propagation delay a collision takes place and the nodes involved have to transition to the BACKOFF state and try again at a later time chosen at random, as shown in Fig 4 10

Node 'x' determines that its RTR was not received correctly by 'z' after a time period equal to the maximum round-trip delay to its neighbours plus turn-around times and processing delays at the nodes, plus the waiting period $\xi$. After sending its RTR, node 'x' listens to the channel for any ongoing transmission. Because of non zero propagation delays, if node 'x' detects carrier immediately after transmitting its RTR, it can conclude that it corresponds to a node other than 'z', which would take a longer time to respond due to its need to delay its data to 'x' to account for turn-around times[1].

The lengths of RTRs and NTRs are the same. The same argument used in to show that the length of an RTS must be longer than the maximum propagation delay between two neighbours to ensure correct collision avoidance can be used to show that RTRs and NTRs must last longer than a maximum propagation delay. In adhoc networks in ISM bands, propagation delays are much smaller compared with any packet that needs to be transmitted

To reduce the probability that the same nodes compete repeatedly for the same receiver at the time of the next RTR, the RTR specifies a back-off-period unit for contention The nodes that must enter the BACKOFF State compute a random time that is a multiple of the back-off-period unit advertised in the RTR. The simplest case consists of computing a random number of back-off-period units using a uniformly distributed random variable from 1 to d, where d is the maximum number of neighbours for a receiver. The simplest back-off-period unit is the time it takes to send a small data packet successfully.

### 4.2.3 RIMA-DP (Dual-purpose Polling).

The collision avoidance strategy described for RIMA-SP can be improved by increasing the probability that data will follow a successful RTR, without violating the rule that data packets should be transmitted only if they are addressed to the polling nodes. A simple way to achieve this with data-driven polling is to make an RTR entry both a request for data from the polled node, and a transmission request for the polling node to send data. The RIMA-DP (receiver-initiated multiple access with dual-purpose polling) protocol does exactly this.

[1] Our analysis assumes 0 turnaround times and 0 processing delays for simplicity.

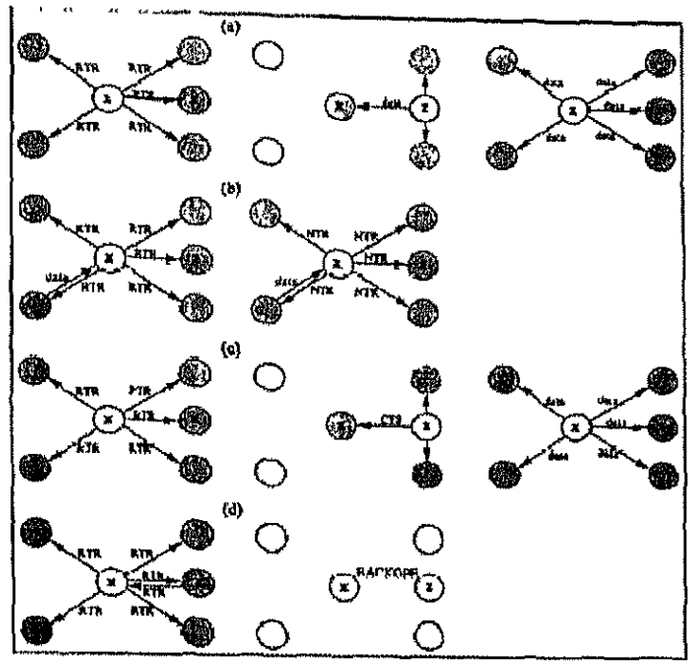


**Figure 4.11** RIMA-DP illustrated [11].

Fig. 4.11 illustrates the modified collision avoidance handshake to permit the polling node to either receive or send data without collisions. As Fig 4.11(a) illustrates, a key benefit of the dual-use polling in RIMA-DP is that both polling and polled nodes can send data in a round of collision avoidance This is possible because the RTR makes all the neighbours of the polling node back-off, and the data from the polled node make all its neighbours back-off, which can then be used by the polling node to send its data.

RIMA-DP gives transmission priority to the polling nodes When a node 'z' is polled by node 'x' and has data for node 'x', 'z' waits $\xi$ seconds before sending a data packet. In contrast, if the polled node does not have data for 'x', it immediately sends a CTS (Clear-To-Send packet) to 'x'. This permits a polling node 'x' exposed to a neighbour sending data to hear part of that neighbour's data packet after sending its RTR; in such a case, node 'x' can send an NTR to the polled node to cancel its RTR. To prevent collisions of data packets, provided that 'z' waits for $\xi > \gamma + 7\tau$ seconds before sending any data after being polled and the length of a CTS is $2\tau$ seconds longer than the length of an RTS As in RIMA-SP, the lengths of RTRs and RTSs are the same

As in RIMA-SP, every node starts in the START State and transitions to the PASSIVE State when it is initialised. If a node 'x' is in the PASSIVE state and senses carrier, it transitions to the REMOTE state to defer to ongoing transmissions. A node in REMOTE State must allow enough time for a complete successful handshake to take place, before attempting to transition from remote state

Any node in PASSIVE State that detects noise in the channel must transition to the BACKOFF State where it must allow sufficient time for complete successful handshakes to occur If node 'x' is in PASSIVE state and obtains an outgoing packet to send to neighbour 'z', it transitions to the RTR state. In the RTR State, node 'x' behaves as in RIMA-SP

If node 'z' receives the RTR correctly and has data for 'x', it waits for $\xi$ seconds before sending a data packet to 'x'. If during the waiting period there is no activity in the channel, node 'z' transitions to the XMIT state, where it transmits a data packet to 'z' Otherwise, 'z' assumes a collision or data transfer to a hidden node and goes to the BACKOFF state. If 'z' has no data for 'x', it sends CTS to 'x' immediately

If node 'x' detects carrier immediately after sending an RTR, it defers its transmission attempt and sends an NTR to the node it polled. The CTS length, which is $\tau$ seconds longer than an RTR, forces polling nodes that send RTRs at about the same time when a polled node sends a CTS to detect carrier from the CTS and stop their attempt to send or receive data Any node other than 'z' receiving the CTS for 'x' transitions to the BACKOFF state. When node 'x' receives the CTS from 'z', it transitions to the XMIT state and transmits a data packet to 'z'

### 4.2.4 RIMA-BP (Broadcast Polling).

Contrary to the prior two approaches, an RTR can be sent to multiple neighbours. We describe a modification of RIMA-SP based on this variant.

A node broadcasts an RTR only when there is a local data packet (data-driven polling). Only after a node has received an invitation, it is allowed to send any data Because a poll broadcast to all the neighbours of a node can cause multiple nodes to attempt sending data to the polling node, an additional control packet is needed to ensure that transmissions that collide last a short period and do not carry user data. Accordingly, a polled node sends a short RTS (Ready-To-Send packet) before sending data. Furthermore, after sending its RTS, the polled node must wait for $\xi$ seconds to allow the polling node to send an NTR when collisions of RTSs occur at the polling node. We call this protocol RIMA-BP (Broadcast Polling).

It can be shown that RIMA-BP provides correct collision avoidance if $\xi = 4\tau$ [11]. Fig 4 12 illustrates the receiver-initiated handshake of RIMA-BP. As it is shown in the figure, the key difference with RIMA-SP is the use of an RTS prior to the transmission of a data packet.

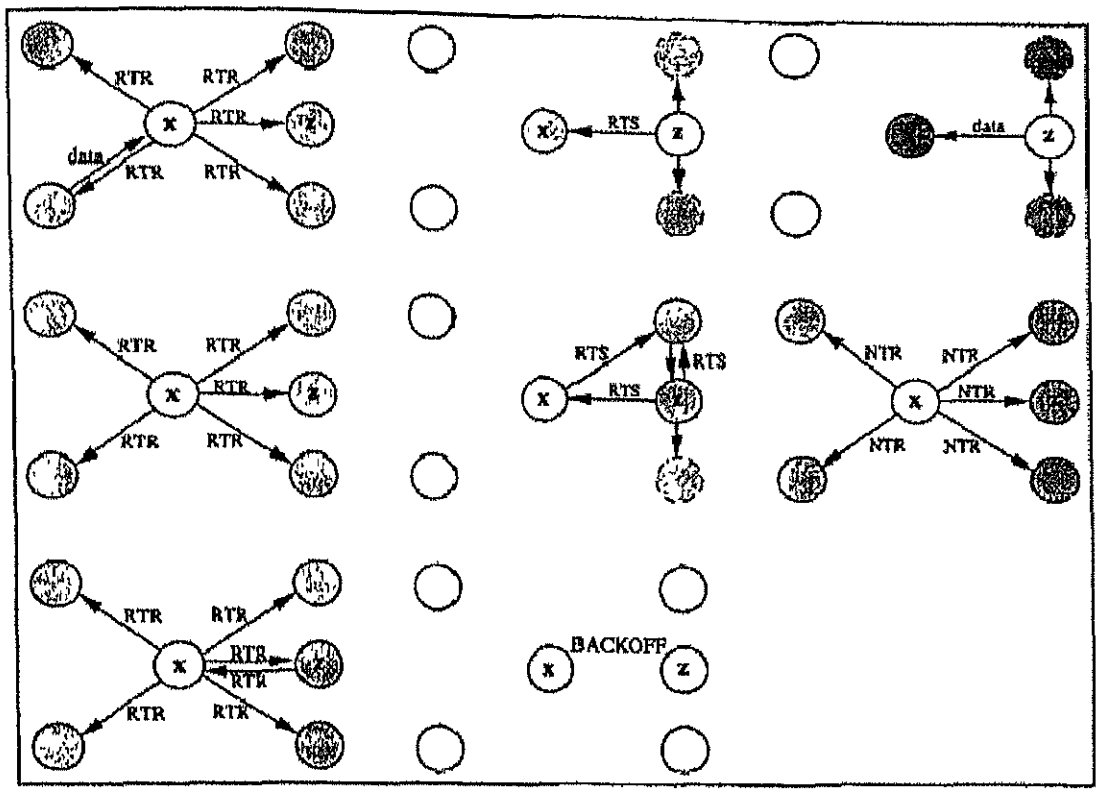



**Figure 4.12** RIMA-BP illustrated [11].

## 4.3 Comparison of Receiver and Sender initiated Protocols.

To compare the various RIMA protocols with MACA, FAMA-NCS, and MACA-BI, we introduce the variables in Table 4.2, and Table 4 3 shows the normalised throughput for the MAC protocols based on those variables In our comparison, we assume a fully-connected network topology with a propagation delay of 1μs; we used 500 byte data packets, a length of 20 bytes for RTRs, CTSs and NTRs for the various RIMA protocols; CTSs of length $\gamma+\tau$ for FAMA-NCS; a channel data rate of 1 Mb/s; and zero preamble and processing overhead for convenience. Figs 4.13,4 14 and 4.15 plot the throughput of MACA, FAMA-NCS, MACA-BI, RIMA-SP, RIMA-DP, and RIMA-BP against the average offered load when the network consists of 5, 10, and 50 nodes, respectively [11]

| |
|---|
| **$a = \tau/\delta$ (normalised propagation delay)** |
| **$b = \gamma/\delta$ (normalised control packets)** |
| **$G = \lambda * \delta$ (offered load, normalised to data packets)** |

**Table 4.2** Normalised variables.

| *Protocol* | *Throughput* |
|---|---|
| MACA | $\frac{1}{a(2b+a)G(b+a+\frac{1}{G}+F')+e^{bG}\left[b+\frac{b}{2}+P'(a-F')\right]+1+\frac{3a}{2}+F'+P'(a-F')}$ where $F' = \frac{e^{aG}-1-aG}{aG(1-e^{-bG})}$ and $P' = \frac{e^{-bG}-e^{-G(a+b)}}{1-e^{-G(a+b)}}$ |
| FAMA-NCS | $\frac{1}{b+4a+1+\frac{1}{G}+e^{aG}(b+4a)}$ |
| MACA-BI | $\frac{1}{1+\frac{1}{G}+a+(b+2a)e^{aG}}$ |
| RIMA-SP | $\frac{\frac{1}{N}}{\frac{1}{N}+\frac{1}{G}+a+(b+2a)e^{aG}}$ |
| RIMA-DP | $\frac{1+\frac{1}{N}}{1+b+2a+\frac{1}{G}+\frac{1}{N}(1+7a)+(b+2a)e^{aG}}$ |
| RIMA-BP | $\frac{(1-\frac{1}{N})^{N-1}}{b+6a+(1-\frac{1}{N})^{N-1}(\frac{1}{G}-2b-5a)+(b+2a+\frac{1}{G})e^{aG}}$ |

**Table 4.3** Throughput of Sender-initiated and Receiver-initiated MAC protocols [11]

The performance attained by RIMA-DP is much better than the performance of the other MAC protocols that provide correct collision avoidance (FAMA-NCS, RIMA-SP, and RIMA-BP). This should be expected, because RIMA-DP permits one or two packets to be sent with each successful handshake, while the other protocols allow just one packet per handshake. As Figs. 4.13 to 4.15 illustrate, the throughput of RIMA-SP degrades as the size of a node neighbourhood increases. Even though our model is only a rough approximation of the impact of the number of neighbours a node has, this illustrates the fact that simple polling is inherently limited compared to dual-use polling, because at light and moderate loads there is a non-zero probability that the polled node has no data to send to the polling node.

It is also interesting to observe that the throughput of RIMA-BP is independent of the number of nodes and is always lower than RIMA-DP There are two reasons for this behaviour: a node receiving a broadcast poll can only transmit packets to the polling node, and multiple responses (RTSs) to the poll are likely to be sent, incurring wasted busy periods.

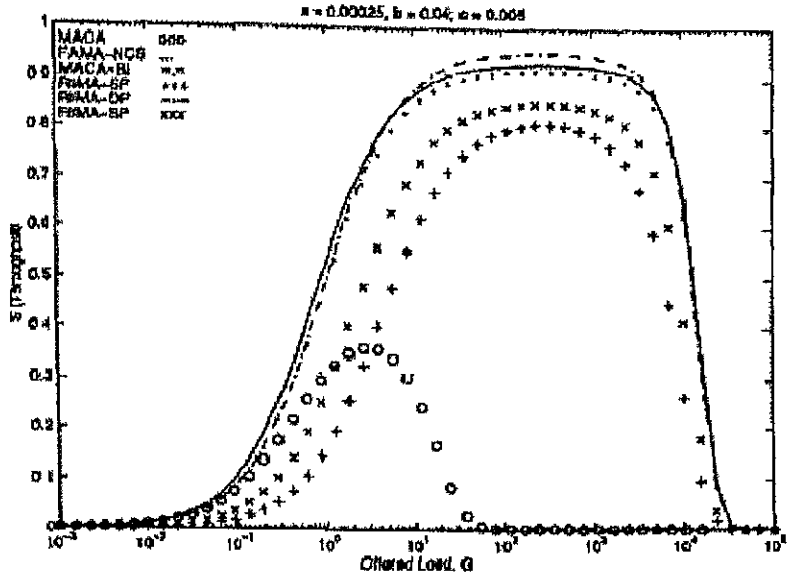



**Figure 4.13** Throughput vs offered load for 1Mbit/sec channel and 500 Byte data packets; network of 5 nodes [11]

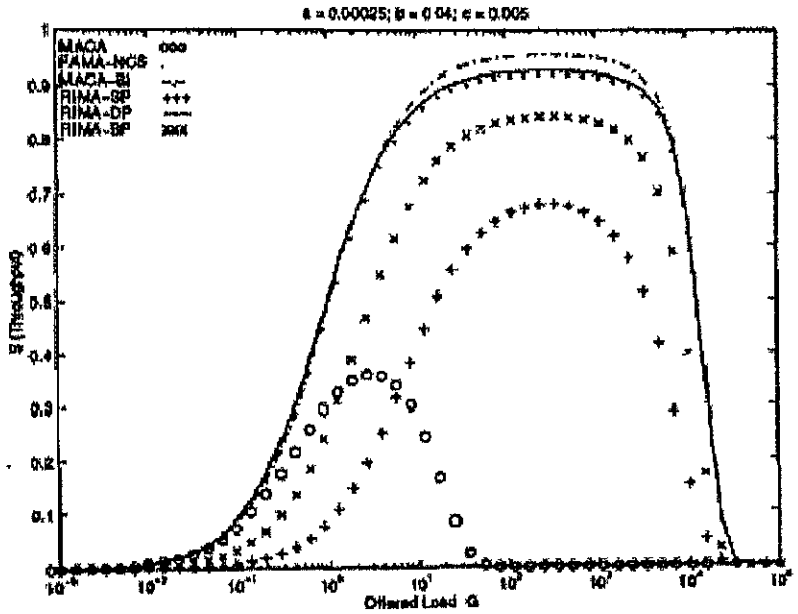



**Figure 4.14** Throughput vs. offered load for 1Mbit/sec channel and 500 Byte data packets; network of 10 nodes [11].

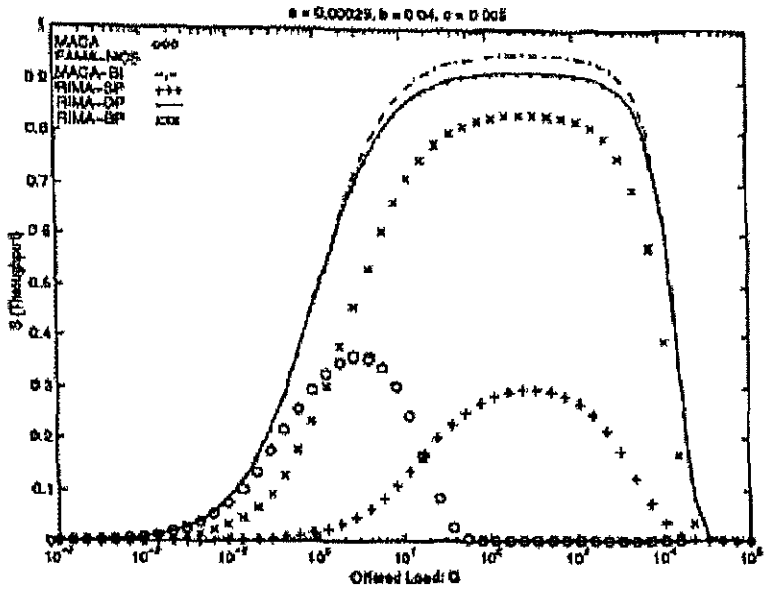



**Figure 4.15** Throughput vs. offered load for 1Mbit/sec channel and 500 Byte data packets; network of 50 nodes [11]

Figs 4.13 to 4.15 also illustrate that carrier sensing is needed to provide high throughput in addition to correct collision avoidance. MACA's poor performance is due to the long durations of busy periods in which collisions occur, which are bounded by a maximum round trip delay and a control packet length with carrier sensing In fairness to MACA and variants of collision avoidance protocols that do not use carrier sensing, it should be emphasised once more that, with the COTS radios available today, carrier sensing is possible only with FHSS (Frequency hop spread spectrum) radios in ISM bands, with which entire packets are sent in a single frequency hop. In contrast, collision avoidance without carrier sensing can be applied to FHSS and

DSSS radios. However, given the performance advantage of collision avoidance using carrier sensing, FHSS radios appear more attractive than DSSS (Direct sequence spread spectrum) radios for ad-hoc networks.

In Figs. 4.13 to 4.15, MACA-BI achieves the maximum throughput among all the protocols The reason for this is that a polled node can transmit a data packet to any node, not just the polling node; however, as we have shown, this should not be done in networks with hidden terminals in which the protocol is meant to operate

To provide a fairer comparison between MACA-BI and RIMA protocols without having to consider a more complex model involving hidden terminals, we can use a heavy-traffic approximation consisting of assuming that a polled node always has data to send to any polling node This approximation is actually not far from reality in large networks in which a node always has packets in its transmission queue meant for different destinations and has to distribute them among its various neighbours With this approximation, the probability that a successful RTR generates two data packets in RIMA-DP is 1, and the probability that an RTR is not answered with data in RIMA-SP is 0; Fig 4.16 shows the corresponding results As could be expected, under the heavy-traffic assumption, RIMA-DP achieves the best throughput under any average load, and RIMA-SP exhibits essentially the same throughput as MACA-BI.

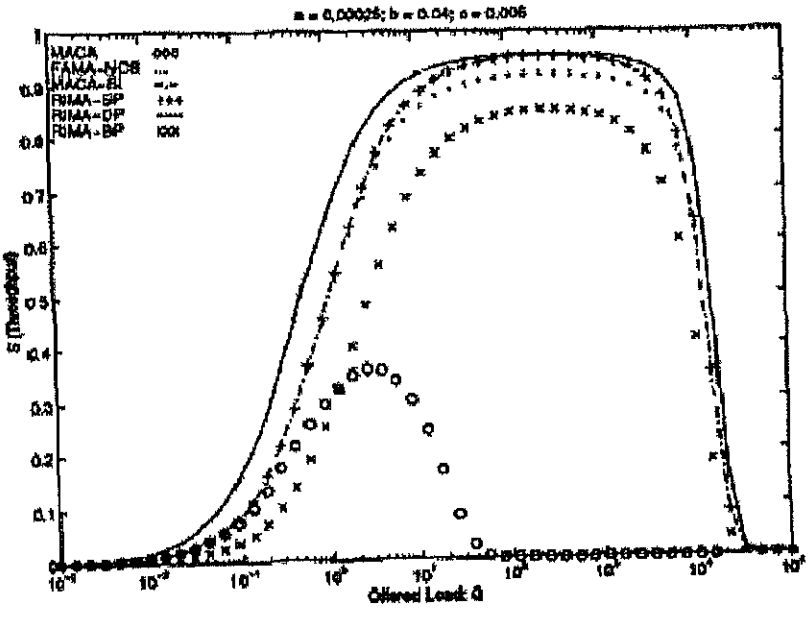



**Figure 4.16** Heavy-traffic approximation. Throughput vs offered load for 1Mbit/sec channel and 500 Byte data packets; network of 50 nodes

It is evident from Figs 4.13 to 4 16 that making collision avoidance a joint effort by sender and receiver, instead of placing all functionality at the sender or the receiver is the best way to avoid collisions while maintaining a high throughput.

# CHAPTER 5
# Modifications In RIMA (Receiver Initiated Multiple Access) Protocols

Many medium-access control (MAC) protocols for wireless networks proposed or implemented to date are based on collision avoidance handshakes between sender and receiver. In the vast majority of these protocols, including the IEEE 802 11 standard, the handshake is sender initiated, in that sender asks the receiver for permission to transmit using a short control packet, and transmits only after the receiver sends a short clear-to-send notification.

We analyse the effect of making the collision-avoidance handshake; receiver initiated and compares the performance of a number of receiver-initiated protocols with the performance of sender-initiated collision avoidance protocols. But in the previous chapter the comparison of various protocols are not fairer, as in Figs 4.13 to 4 15, MACA-BI indicates the higher throughput as compared to the other RIMA protocols and its various versions, while in all the versions of RIMA, it has shown RIMA-DP as the best protocol among the receiver initiated policy. The heavy traffic approximation shown in Fig. 4 16 does not match the requirements of the multi-hop networks So the comparison of MACA-BI with RIMA protocols do not fit well [11] As from the discussion in the previous chapter, especially Figs 4 13 to 4.16, it is clear that as we keep on increasing the number of nodes; the throughput variation in RIMA-BP is less as compared to other RIMA protocols In this chapter, we have tried to show more variants of RIMA-BP protocols and its comparison with the original RIMA-BP protocol. By considering some realistic assumptions, we have tried to show its effect on the throughput performance of various receiver initiated protocols.

## 5.1 Modified RIMA-BP Protocol

In RIMA-BP, an RTR can be sent to a broadcast address by polling node to the polled nodes and multiple neighbours can receive and decode the packet at the same time. If any of the polled nodes have packet to transmit, then prior to data transmission, it will send RTS, to indicate that it has data for polling node But in case of broadcasting, multiple nodes can send RTS leading to collisions.

To avoid these collisions, we have introduced the concept of *random waiting time* which not only avoids the collisions among RTSs, but also improves the throughput with the increase of fairness

policy which is one of the issue in the design of MAC layer protocol in AD-HOC networks. It also leads to the reduction of control overhead, which is having manifold advantages. Because in order to carry out actual data transmission in this modified version of RIMA-BP, less number of handshakes are there, hence we can neglect the effect of hardware transmit to receive time or turn around time ε, in the through-put analysis of this protocol [10]

In this modified RIMA-BP, polling node will send RTR to broadcast address and if multiple polled nodes have data then prior to transmit they will choose randomly the waiting time and after waiting for that time, first it will sense the carrier by using the non-persistent policy (as it was shown in prior protocols that non-persistent gives maximum throughput among all persistent scheme) and if it will find channel free then it will transmit

This whole handshake is given in Fig 5 1. In this, polling node X will send RTR to its

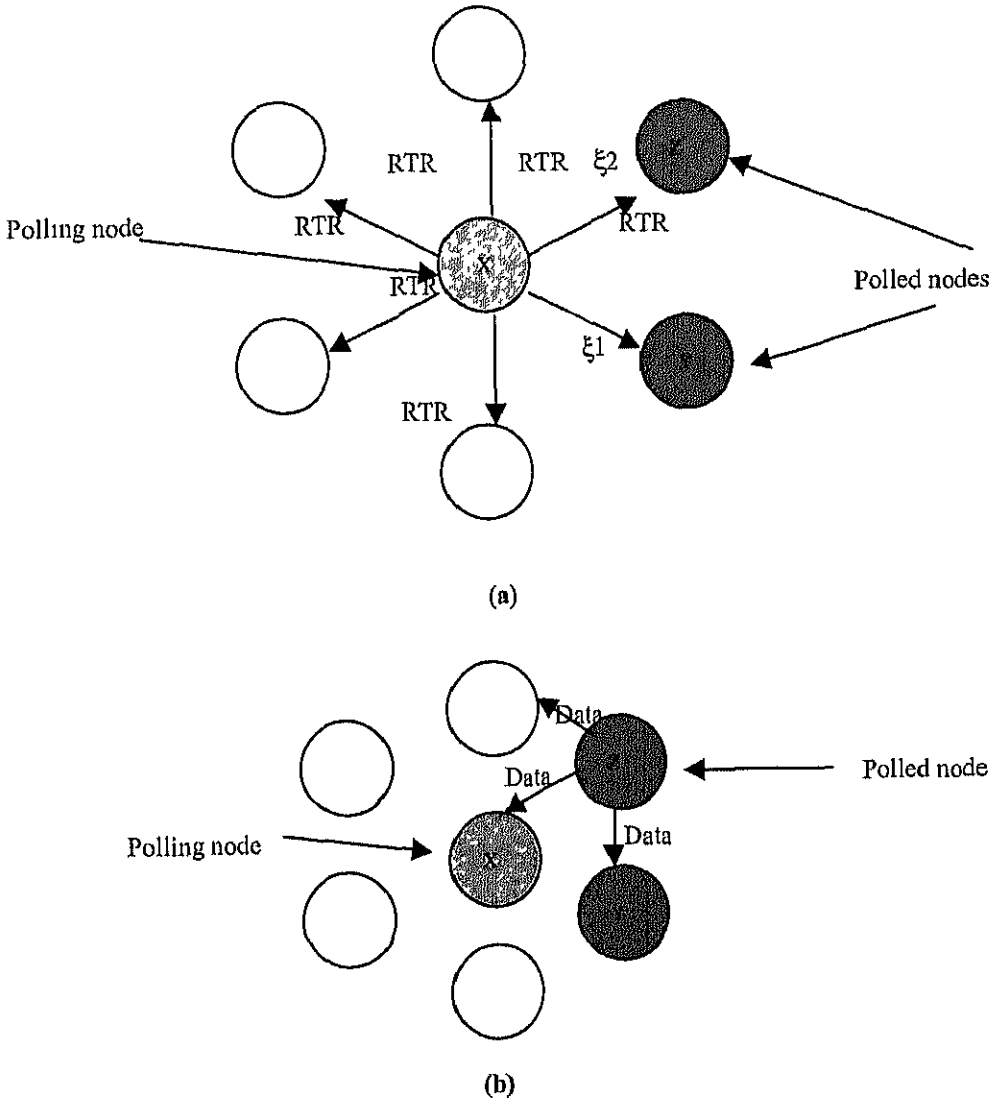



**Figure 5.1** (a) Polling node transmitting (b) Polled node transmitting.

neighbours If both polled nodes Y and Z have data to polling node X, then prior to transmission they will select waiting times randomly (they can select their waiting time on the basis of MAC addresses to avoid full contention) say $\xi 1$ and $\xi 2$ (if $\xi 1 > \xi 2$) respectively and after waiting for this time first they will sense the channel, if it is free then they will transmit otherwise they will Backoff There is less probability that both nodes will choose the same waiting time In this all nodes will execute binary exponential backoff algorithm during Backoff State

In this protocol, to make the correct data transfer either of the two (polled and polling node) will trace the five states which are START, PASSIVE, WAITING, BACKOFF and TRANSMIT [11]

### 5.1.1 Approximate throughput analysis.

We analyse the throughput of receiver-initiated protocols using the model first introduced by Kleinrock and Tobagi [5] for CSMA as discussed in chapter 3 and used subsequently to analyses MACA-BI [10] and RIMA [11] and several others collision avoidance protocols. According to this model, the following assumptions are made.

1 There are $N$ nodes in the fully connected network.

2 A single unslotted channel is used for all packets, and the channel introduces no error

3. There is no capture or fading in the channel.

4. All nodes can detect collisions perfectly

5. The maximum end-to-end propagation time in the channel between any two nodes is $\tau$ seconds

6. The size for a data packet is $\delta$ seconds, the size of an RTR and an ACK is $\gamma$ seconds and the waiting time is $\xi$ seconds.

7. Hardware transmit to receive transition time is zero, furthermore $2\tau < \gamma \leq \delta < \infty$.

Fully connected network means every node can hear the transmission of each other The present analysis includes the overhead incurred by the ACKs needed to inform the sender of the correct reception of a data packet. The probability that the packet is addressed to the polling node is $1/N$. Furthermore, we assume that each node sends its RTR according to a Poisson distribution with a mean rate of $\lambda/N$, and that (when applicable) the polling node chooses the recipient of RTR with equal probability

Because the arrival of RTRs to the channel is Poisson, the average channel utilisation is

$$S = \frac{\overline{U}}{\overline{B} + \overline{I}} \qquad (5.1)$$

where $\bar{B}$ is the expected duration of a busy period, defined to be a period of time during which the channel is being utilised; $\bar{I}$ is the expected duration of an idle period, defined as the time interval between two consecutive busy periods, and $\bar{U}$ is the time during a busy period that the channel is used for transmitting user data successfully.

Given our independence assumptions, the probability of success, $P_s$, equals the probability with which an RTR is transmitted successfully. Because all nodes are connected, an RTR from node w is successful if there are no other RTRs transmitted within $\tau$ seconds, where $\tau$ is the time needed for all the nodes connected to detect the carrier signal. After this vulnerability period of $\tau$ seconds, all the nodes detect the carrier and act appropriately. Because the arrivals of RTRs to the channel follow the Poisson distribution with rate $\lambda$, we can write

$$P_s = e^{-\lambda\tau} \tag{5.2}$$

The duration of every successful busy period is $(\gamma + \delta + \xi + 2\tau)$, and the first and the last packet of the busy period is the successful packet of the period

The average duration of any busy period always consists of at least an RTR and the associated propagation delay (i.e., $\gamma + \tau$) plus the average time between the first and the last RTR of the busy period, which we denote by $\bar{Y}$ and is the same as in CSMA [5], i.e ,

$$\bar{Y} = \tau - \frac{\left(1 - e^{-\lambda\tau}\right)}{\lambda} \tag{5.3}$$

There are two ways in which a busy period can be unsuccessful, i.e., contain no data packet. First, the RTRs sent in the busy period may collide with one another, which occurs with probability $1 - e^{-\lambda\tau}$ because all nodes can hear one another. A busy period can also fail if a single RTR is sent in the clear but none of the polled nodes has a packet to send to the polling node; the probability with which this scenario takes place is equal to:

$$P_{F1} = e^{-\lambda\tau}\left(1 - \frac{1}{N}\right)^{N-1} \tag{5.4}$$

With the probability $P_s$, the busy period also includes a collision-avoidance waiting time of the polled node, the data packet from the polled node, the ACK from the polling node, plus the associated propagation delays. With probability $P_{F1}$, the busy period also contains a waiting time of $2\tau$ after which polling node detects no data from polled node. Accordingly, the duration of an average busy period is

$$\overline{B} = \gamma + 2\tau - \frac{1-e^{-\lambda\tau}}{\lambda} + e^{-\lambda\tau}\left(1-\frac{1}{N}\right)^{N-1}(2\tau) + e^{-\lambda\tau}(\gamma+\delta+\xi+2\tau) \tag{5.5}$$

In addition, the channel is idle for a time period equal to the inter arrival rate, so $\overline{I} = 1/\lambda$ The average utilisation time at node w is the proportion of time in which useful data are sent Consequently,

$$\overline{U} = \delta \cdot e^{-\lambda\tau} \tag{5.6}$$

Substituting the equations for $\overline{B}$, $\overline{U}$ and $\overline{I}$ into equation (5 1), we obtain

$$S = \frac{\delta}{(\gamma+2\tau)e^{\lambda\tau} + \frac{1}{\lambda} + \left(1-\frac{1}{N}\right)^{N-1}(2\tau) + (\gamma+\delta+\xi+2\tau)} \tag{5.7}$$

## 5.1.2 Numerical results.

To compare the RIMA-BP with the modified RIMA-BP protocol, and other RIMA protocols in the previous chapter we introduce the variables in the table 5 1

| |
|---|
| **a = τ/δ (normalised propagation delay)** |
| **b = γ/δ (normalised control packets)** |
| **G = λ * δ (offered load, normalised to data packets)** |

**Table 5.1** Normalised variables.

In our comparison, we assume a fully-connected network topology with a propagation delay of 1μs; we used 500 byte data packets; a length of 20 bytes for RTRs, CTSs and NTRs for the various RIMA protocols; a channel data rate of 1 Mb/s; and zero preamble and processing overhead for convenience. Figs 5.2, 5.3 and 5.4 plot the throughput of RIMA-BP original, RIMA-BP modified, RIMA-SP and RIMA-DP against the average offered load when the network consists of 5, 10, and 50 nodes, respectively.

From these graphs it is clear that there is improvement in the throughput in RIMA-BP modified as compared to RIMA-BP original and it is nearly comparable to RIMA-DP which in the previous chapter was considered to be the best protocol among the receiver initiated policy.

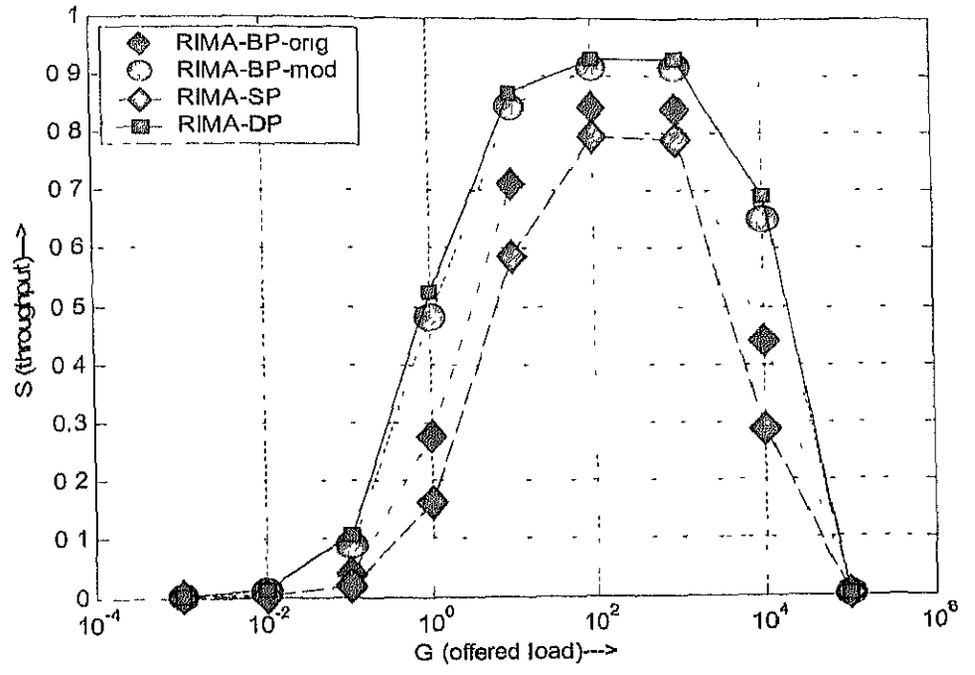



**Figure 5.2** Throughput versus offered load for 1Mbit/s channel and 500 Bytes data packets: Network of 5 nodes

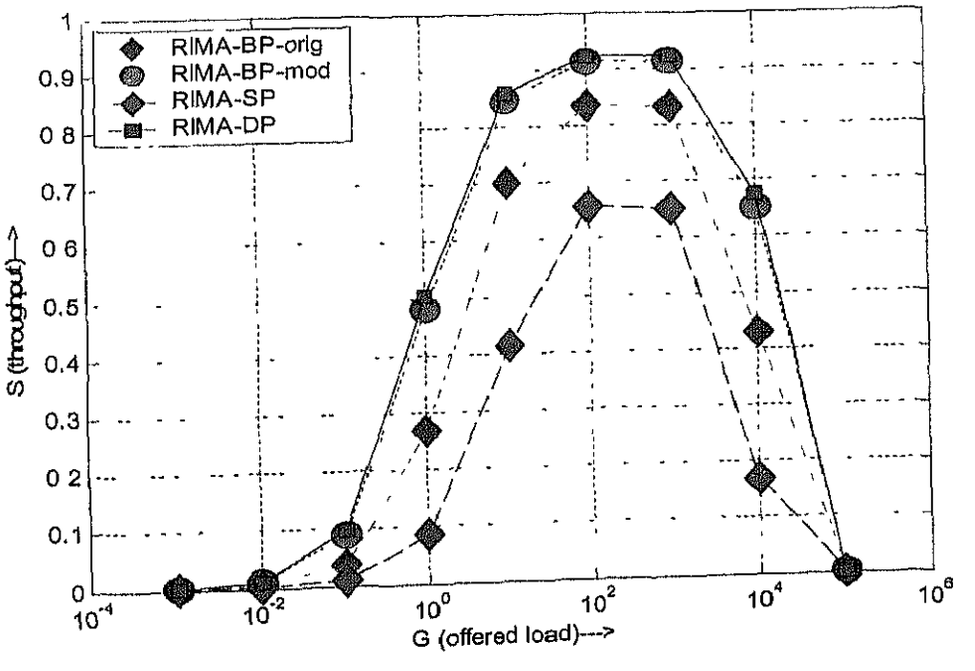



**Figure 5.3** Throughput versus offered load for 1 Mbit/s channel and 500 Bytes data packets. Network of 10 nodes

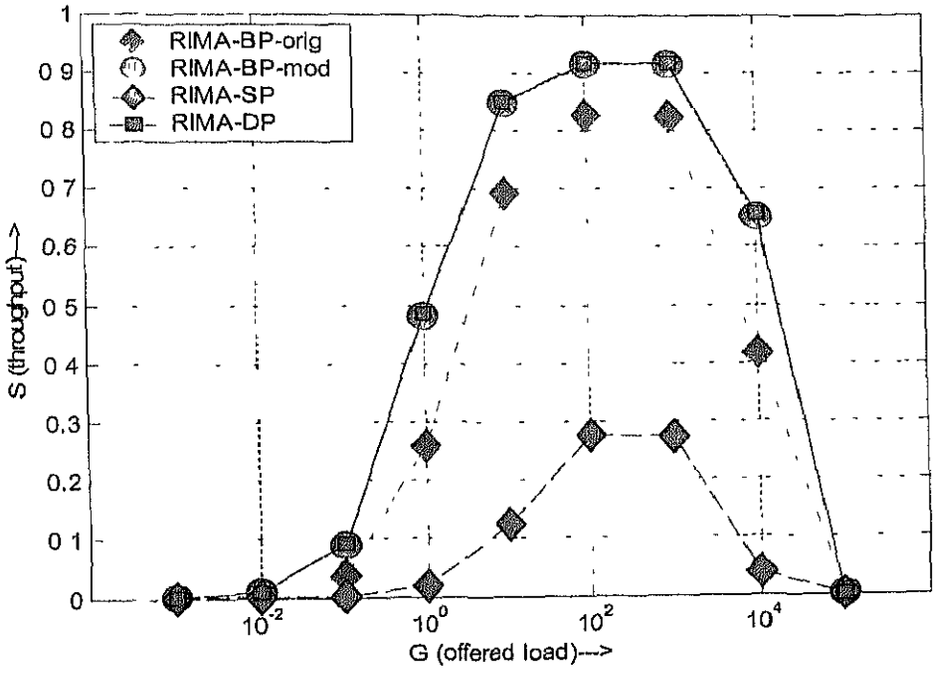



**Figure 5.4** Throughput versus offered load for 1 Mbit/s channel and 500 Bytes data packets: Network of 50 nodes.

The worst case improvement (i.e., for N=50) in the throughput of RIMA-BP modified as compared to the RIMA-BP original is nearly around 11% as shown in Fig. 5.4, which is quite considerable. Also for N=50, the throughput of RIMA-BP modified is same as that of RIMA-DP and for less number of nodes there is minor difference in the throughput.

But this much of throughput in the RIMA-BP modified is with less number of handshakes between polling and polled nodes as compared to RIMA-DP, which is having more handshakes to initiate the data transfer. If the channel rate is high, then the protocol which is having more handshakes, the effect of hardware transmit to receive time becomes prominent. Hence, at high channel rates we can not neglect the effect of hardware transmit to receive time on RIMA-DP protocol, but in case of RIMA-BP we can neglect its effect as less number of handshakes are there.

### 5.1.3 Prediction of Random waiting time in RIMA-BP modified.

Now the question arises, how will we predict the range of random waiting time in RIMA-BP modified? Now proceeding with the previous analysis, through simulation we have tried to predict the

range of random waiting time and the optimum range of the waiting time for which the throughput is maximum. It is clear from Fig. 5.5, that within 2% tolerance in the throughput of RIMA-BP modified, the range of waiting time comes out to be from 1μsec to 100 μsec, under worst condition i.e., when $N$=50

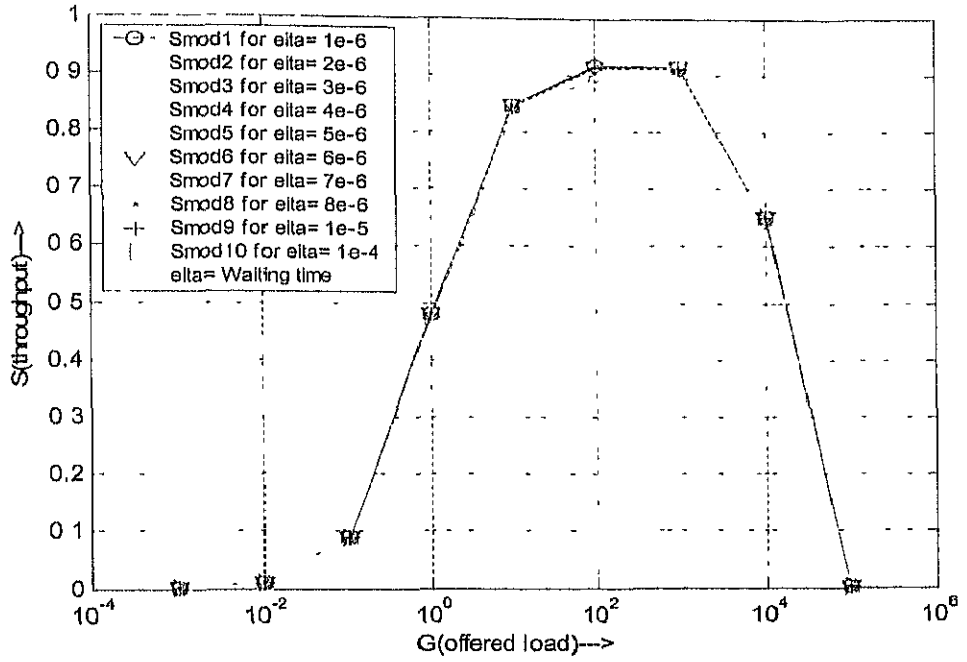



**Figure 5.5** Plot of throughput versus offered load for RIMA-BP protocol with varying value of $\xi$ (Network of nodes 50, channel rate 1Mbps).

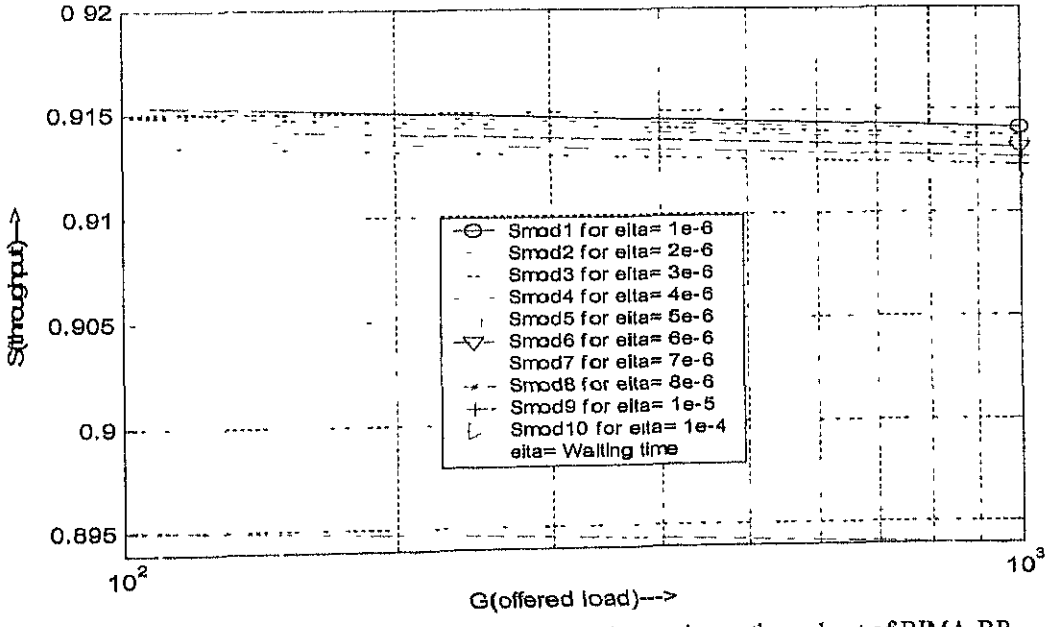



**Figure 5.6** Extended view of above Fig. 5.5 for the maximum throughput of RIMA-BP modified protocol (i.e., G = $10^2$ to $10^3$).

From the previous figures, it is clear that the range of waiting time under worst case lies between 1 μsec to 100 μsec, but to know about the optimum value of waiting time and its range, the clearer picture is given in Fig 5.7 In this plot of throughput versus ξ (waiting time) for only that values of offered load for which throughput is maximum gives the optimum waiting time around 10 μsec and its range lies from 1 μsec to 20 μsec which gives very less variation in the throughput of RIMA-BP modified

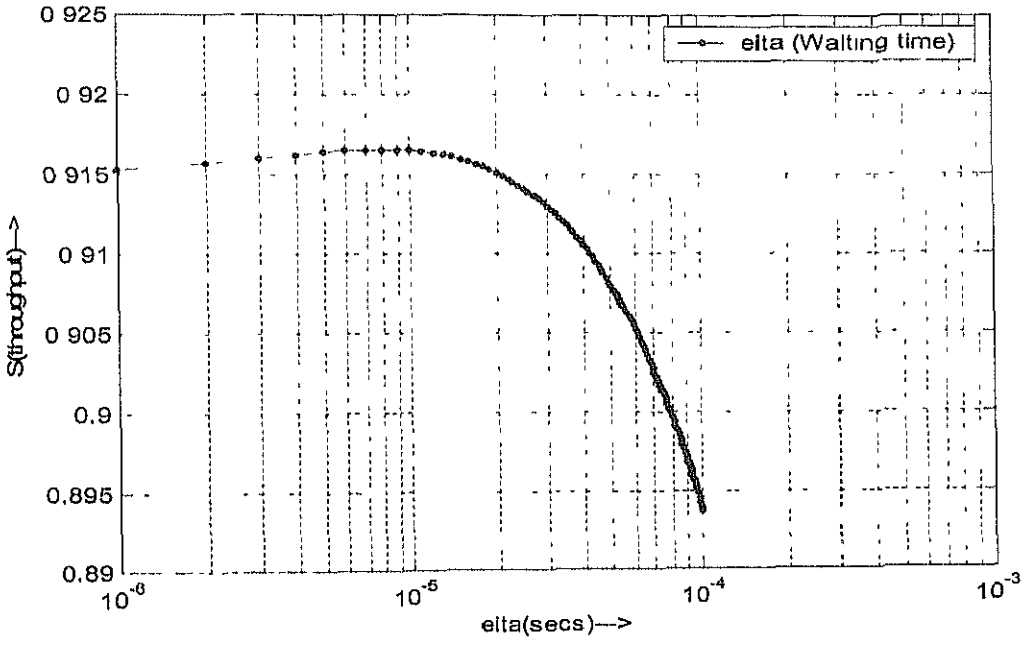



**Figure 5.7** Throughput versus waiting time only for that value of G for which S is maximum

## 5.2 Effect of hardware TX to RX time on various Receiver Initiated Protocols.

If the number of handshakes are more in the protocol, then with each additional pass in the handshake contributes one TX-RX turn around time plus preamble bits (for synchronisation) [10], control bits (e.g., source-destination information) and checksum bits. This overhead clearly reduces the throughput.

According to the standard proposed in [15], the TX-RX turn-around time should be less than 25μs (including radio transients, operating system delays and energy detection) Moreover, every transmission should be delayed by the TX to RX turn-around time (i e , upped 25μs) to give a chance to

the previous transmitter to switch to receive mode The higher the channel speed, the higher the turn around time overhead in term of bits. Thus, turn around will play a key role in future high-speed wireless LANs.

As we have already seen that in case of RIMA-BP modified the number of handshakes are less; hence even if the channel rate is high its throughput will not be much effected But we have tried to show the effect of hardware TX-to-RX turn around time explicitly on throughput of RIMA-SP, RIMA-DP, RIMA-BP and MACA-BI protocols.

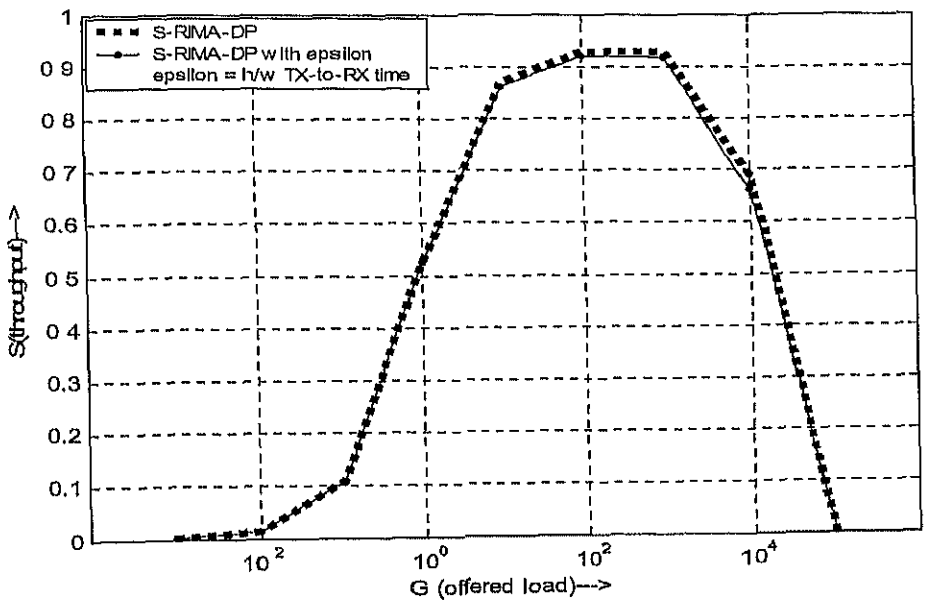



**Figure 5.8** Effect of h/w TX-to-RX time (ξ) on the throughput of RIMA-DP for N=50

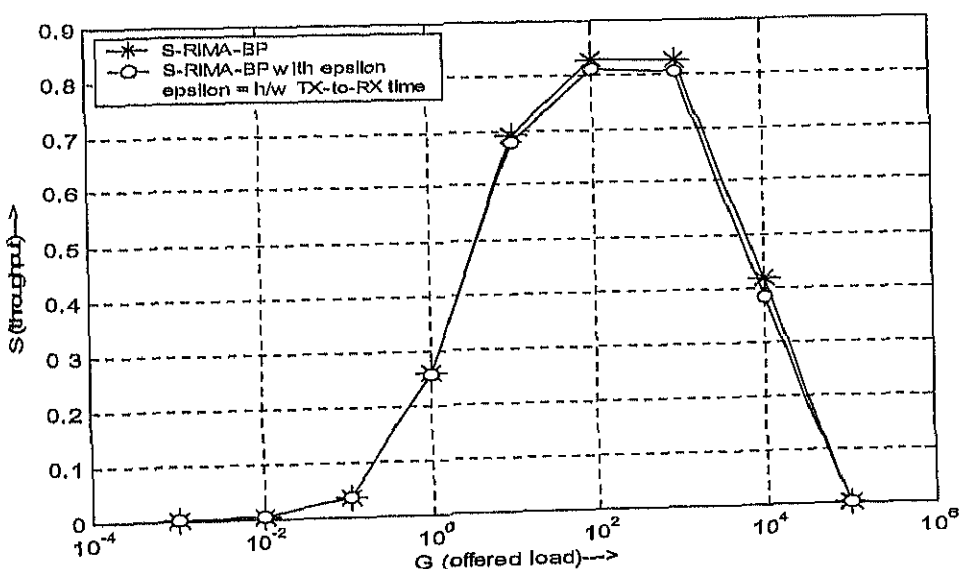



**Figure 5.9** Effect of h/w TX-to-RX time (ξ) on the throughput of RIMA-BP original for N=50

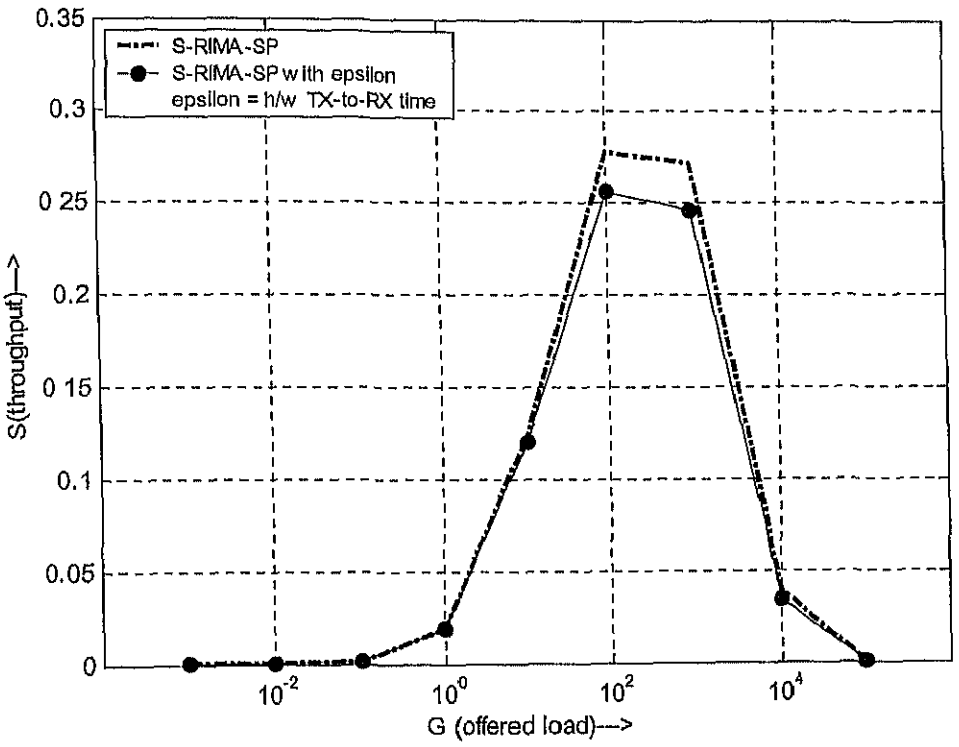



**Figure 5.10** Effect of h/w TX-to-RX time (ξ) on the throughput of RIMA-SP for N=50

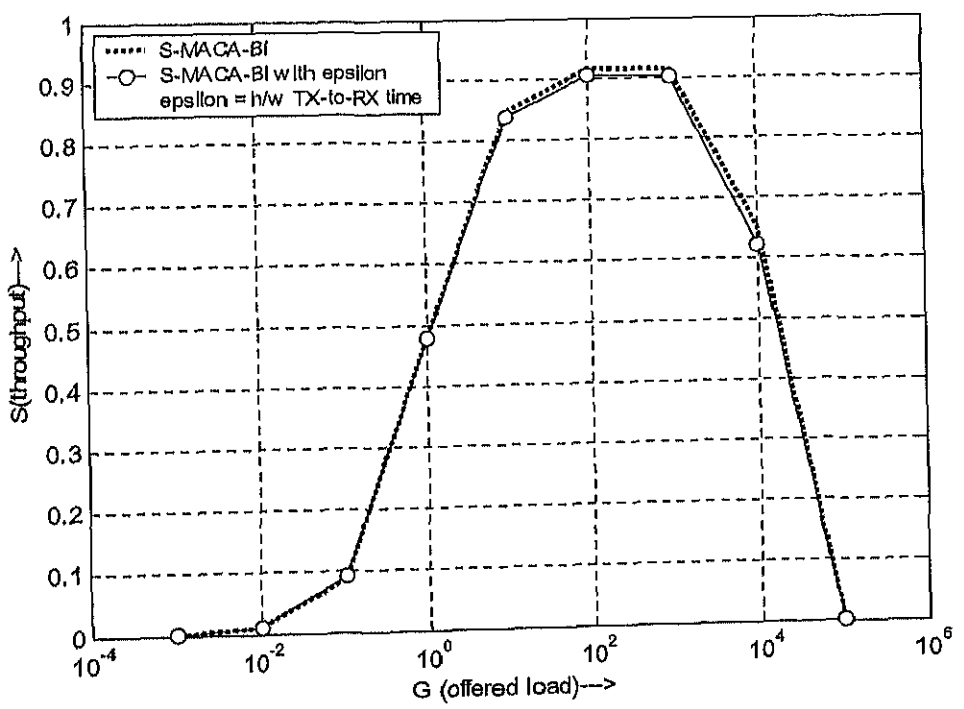



**Figure 5.11** Effect of h/w TX-to-RX time (ξ) on the throughput of MACA-BI.

From the above figures it is clear that under worst condition (i e. N=50), the effect of $\xi$ on RIMA-SP is more as compared to other receiver initiated protocols But above effect is observed when channel rate is only 1Mbps, if it will be high then situation will be more worst From Fig 5 10 it is clear that reduction in the throughput is around 7%. While in RIMA-BP original reduction observed is around 2% which can increase with the increase in the channel data rate But in our proposed scheme of channel access having the concept of random waiting time, is free from this impairment, and it performs as good as RIMA-DP under worst condition, which in the previous chapter was considered to be the best receiver initiated approach

# Conclusions and Results

The variant of RIMA-BP protocol discussed in this thesis comes out to be fruitful in comparison of RIMA-DP, which is the best protocol among the receiver-initiated policy proposed by J.J. Garcia-Luna-Aceves et all. The improvement in the throughput in RIMA-BP modified is nearly 11% over the throughput of RIMA-BP, which is bountiful. Within 2% range of throughput tolerance, the range of waiting time is quite reasonable, which is also quite clear from the simulation results presented in the previous chapter. We can also assign priority in the selection of waiting time by nodes on the basis of high MAC addresses to avoid full contention. Also in this protocol, not only the effect of hardware transmit to receive time is negligible (As there are less number of handshakes to carry out the data transfer), but also there is reduction in the control overhead (This is because of the removal of RTS in response of RTR sent by the polling node). Hence, we can use this protocol for future high-speed wireless AD-HOC LANs.

# Further Studies

This is an inchoate study composition towards the design of MAC layer protocol for AD-HOC wireless LANs; there is an ample scope to study and develop it further. The following areas are proposed to study and extend this exposition up to bench mark level:

(a) Extension of all MAC layer protocol to multi-hop networks.

(b) Effect of channel fading on the throughput performance.

(c) Receiver initiated protocols based on schedules is an area of future research for wireless networks.

(d) Study on distributed queuing approach.

(e) Study on QOS based routing and topology decision in AD-HOC networks.

(f) Delay performance calculations.

(g) Power management in AD-HOC networks.

(h) Timing synchronisation in slotted AD-HOC networks.

(i) Fairness issues in AD-HOC networks.

# REFERENCES


1. Bob O'Hara and Al Petrick, "*IEEE 802.11 Handbook A Designer's Companion*", Standard Information Network, pp 1-5.

2 Dimtri Bertsekas, Robert Gallager,"*Data networks* ",2nd ed , *PHI* private Ltd , 2001, pp. 271-273

3 Andrew S. Tanenbaum,"*Computer networks* ", 3rd ed , *PHI* private Ltd , 2000, pp 262-265 and 241-249

4 Simon Haykin,"*Communication Systems 3/e*", 3rd ed ,John Wiley & Sons, pp 733-735

5 L Kleinrock and F A. Tobagi, Packet switching in radio channels Part 1 – Carrier sense multiple-access modes and their throughput delay characteristics, IEEE Transactions on Communications 23(12) (December 1975) 1400-1416.

6. P. Karn, MACA – a new channel access method for packet radio, in. *proceedings of ARRL/CRRL Amateure Radio 9th Computer Networking Conference*, New York (April 1990).

7. V. Bharghavan, A.Demers, S.Shenker, and L Zhang,MACAW: A media access protocol for wireless LAN's, in: *Proceedings of ACM SIGCOMM*, London, UK (August 1994) pp 212-225.

8. Juha Heiskala and John Terry, "*OFDM wireless LANs. A theoratical and Practical Guide*", pp. 215-256.

9. C.L. Fullmer, and J.J. Garcia-Luna-Aceves, Floor acquisition multiple access for packet radio networks, in: *Proceedings of ACM SIGCOMM*, Cambridge, MA (september 1995).

10. F. Talucci, M Gerla and L. Fratta, MACA-BI (MACA by invitation) – a receiver oriented access protocol for wireless multihop networks, in: *Proceedings of IEEE PIMRC* (1997)

11. J.J. Garcia-Luna-Aceves and Asimakis Tzamaloukas, Receiver initiated collison avoidance in wireless networks, Wireless Networks 8, pp. 249-263, 2002.

12. J. Weinmiller, M. Schlager, A. Festag, A. Wolisz. *Performance study of access control in wireless LANs —IEEE 802.11*. Advances in wireless communications, pp 219-226, 1998

13. Bernard H. Walke, "*Mobile Radio Networks; Networking and protocols*", John Wiley and Sons, pp 699-704.

14. http://www.ietf.org.

15. Wireless LAN medium access control (MAC) and physical layer (PHY) specifications P802.11 D2.0 Unapproved IEEE Draft Standard, July 1995.

143485